

SyncWare News

The journal for technical
applications of T/S
computers

Volume 4 Number 4

Mar.-Apr. '87

SWN On Line

No, we haven't started a bulletin board. Nor have we been bought out by CompuServe. We have gotten a little more involved in the telecommunications end of T/S computing though. Look for us most Wednesday nights in the Timex/Sinclair SIG on CompuServe. Just GO CLUB and meet with us, and a bunch of other good folk, from about 10:00 to 11:00 pm EST. We also accept EASYPLEX mail at 72466,3324. We'll be checking our 'mailbox' about once weekly if you care to leave us a note.

As a convenience to our authors, we're looking into the possibility of providing a means of uploading articles to one of our computers at the Ohio editorial offices. As we ponder the feasibility of this, we welcome your comments and suggestions.

Computerfest Update

May 2-3 is the date and Indianapolis, Indiana is the place to be for the 1987 session of the T/S Computer Fest! According to Executive Director Paul Holmgren, vendor registration is already at last year's level with more reservations coming in! The Holiday Inn North (1-317-872-9790) is the official host of this year's event. They are providing 5000 square feet of exhibit space. Registration for the Fest is currently \$6.00 per single or \$9.00 per family. To register, contact Paul Holmgren, 5231 Wilton Wood Ct., Indianapolis, IN 46254. For information, please send a SASE.

Contest Deadline Nears

The Second (sort of) Annual Programmer's Contest deadline is rapidly approaching. As of January 31, only a single entry had been submitted. It sure would be a shame for a fellow with a single entry to walk away with all the prizes! Maybe the holiday season, the flu, or just mid-winter depression has kept you from your keyboard. If so, we're going to help you out. The contest deadline has been extended to April 15, 1987. Now you've got something to look forward to other than just paying your taxes! For more details, and the contest rules, see the display ad inside.

SILICON MOUNTAIN COMPUTERS announces TRUE HIGH RESOLUTION SOFTWARE for the ZX81/TS1000. You read it right! Without any expensive hardware add-ons, your computer can now run software that even its designers never dreamed possible. Thanks to an amazing discovery by Wilf Rieger, and innovative programming by Gregory Harder and Fred Nachbaur, you no longer have to suffer the "low res blues." Multiple character sets, 256x192 graphics, 64-column screens, UDG's, even SPRITES are now available for your computer!

NO computer modifications are required. If you have a ZX81, TS1000, or TS1500, with a 16K (or larger) RAM pack, plus an 8K static RAM board, you already have all it takes to run this remarkable software. Suitable static RAMs include the popular "Hunter" board, or similar designs (see SynWare News vol. 4 no. 1 for one such project costing under \$10). Alternately, consider our 8K "SCRAM" board (described below).

All prices include shipping in USA and Canada. Foreign orders: please add \$5 for air shipping. CDS\$ accepted at par from Canadian customers. Write for catalog of other available software. At SILICON MOUNTAIN COMPUTERS, the ZX81 family of computers is our ONLY specialty. Our goal is to develop the most progressive software ever created for these machines. We feel that the software listed below procees these machines into mainstream computing; we think that you'll agree.

With the loss of the "Hunter" board from the market, we saw the need for an improved functional equivalent, at a lower price! This board works with ZX81/TS1000 or TS1500, has on-board battery back-up protection, and supports all of our high-res software with no modifications. It can be mapped in 0-8K (ROM overlay), 8-16K (normal operation), even 16-24K or 24-32K.

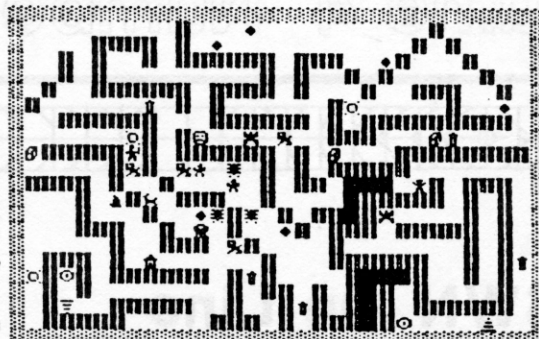
- * DIP switch to deselect 2K blocks
- * Board enable switch
- * Write-protect switch
- * RESET switch easily installed (optional)
- * Very low power drain
- * Feed-through connector
- * FULLY ASSEMBLED! Just plug it in.
- * Use with other machine-code software

DUNGEON OF YMIR

This D&D-style graphic adventure game is the most challenging, complex and spectacular entertainment software ever written for the ZX81 family of computers. 9 levels, 124 monsters (16 types), 5 spells, 15 objects. VERY addictive, VERY challenging, VERY impressive graphics. 100% machine-code, yet takes up all available memory in 15K. One reviewer calls it "5-star software" and "a must have." We know that you'll agree.

***** NEW LOWER PRICE *****

PRICE: \$19.95 including shipping



A King Miner
Lit= 14 | Exp=4640 | Co 0 | Tel | Sh2 | Br2 | R0 | Wx5

```

    NOO      CLEAR  CLS      RETURN  PLOT      UNPLOT  COPY      SAVE
    LOAD     PHASE  LIST     LLIST     PRINT     REVERSE  POINT     LOCATE
ADDRESS      DRAG            WINDOW  WINDOWL  WINDOWH  WDG
CIRCLE              SPAN W-RES    WINDOW
RECTANGLE       UNCIRCLE  WIRCIR
TRIANGLE    INVERT WINDOWL WINDOWH WINDOWL WINDOWH WDG  SPRITED  SPRITE
SPRITED     SPRITEN SPRITES  SETMODE  SAVE-V  LOAD-V  SAVE-S  LOAD-S

```

by GREGORY C. HARDER
and
FRED NACHBAUR

 URX16 Core routine developed by WILF RIGTER

Copyright, 1986

WIC (↑, ↓, ←, →, ~, N) (&

SRAM HI*RES EXTENDED BASIC is the flagship of our new line of software. With this remarkable package YOU can write high-resolution applications... ENTIRELY IN BASIC! While using only 4K of memory, SRAM HI*RES adds 38 new hi-resolution commands. If you know how to program in Sinclair BASIC, you will find SRAM HI*RES easy to learn and use. A revolutionary syntax system allows ANY variables or expressions to be used in your commands. No REMs to pass parameters! No POKES! A singleUSR call is used for ALL commands! Most commands can be chained into MULTIPLE STATEMENT LINES! We even included a fast (8.5x normal) set of tape routines! The most reliable tape system ever written. Other features:

- * Tape system event printer: Other features
- * Three 32-column PRINT modes
- * Lower-case and new symbols
- * 64-column PRINT mode
- * 128 User-defined characters
- * Scroll WINDOWS any direction...
 - o a pixel at a time!
- * Up to 32 TRUE sprites! Speed adjustable.
- * Invert windows or entire screen
- * Software-only video reverse
- * PLOT, DRAW, CIRCLE, RECTANGLE, TRIANGLE
- * TS2040 printer supported
- * Much more!

YEAR-AT-A-GLANCE

This calendar/appointment book program demonstrates the power you have using SRAM HI*RES Extended BASIC. Enter, update, delete, list, print messages and reminders for any day of any year 1800-7099. Program structure will remind you of much larger machines which shall remain nameless...

PRICE: \$9.95 including shipping.

*** SEE SPECIAL OFFER BELOW ***

[illegible]

01 B2-B3 E7-E5
02 C1-A3 B8-C6
03 A3-E1 E3-F8
04 D1-H3 G8-F8
05 D2-D4 C6-D4
06 E2-E4 F6-E4
07 R1-A3 E4-C3
08 D1-D3 C3-D5
09 C2-C3 D8-F6
10 D3-C4 C7-C6
11 G2-G4 D4-E3
CHECK
12 E1-E2 B7-B5
13 A3-B5 D5-B6
14 C4-C5 F8-E8
15 B5-B6 E8-E7
16 D6-E4 D2-D6
17

If you purchase our "SCRAM" NVM board and SRAM HI*RES EXTENDED BASIC and mention this ad, you'll get a FREE copy of YEAR-AT-A-GLANCE. Offer good only until July 1, 1987.

C-12, Mtn. Stn. Group Box, Nelson BC V1L 5P1, CANADA (604)352-1668

FOR YOUR SUPPORT

Paul Holmgren, 5231 Wilton Wood Ct., Indianapolis, IN 46254, asks you to try it before you buy! Where else can you get this kind of deal? I.S.T.U.G. is selling copies of their BBS program. You get the program plus full directions to set up your own BBS on your TS2068. To sample the system, call (317) 898-3903 [7,1,E: 24 hrs.]. Too many features to mention here. Said to run 99.99% trouble free. Price \$16.00.

Glenn Technics, Box 2760, Anaheim, CA 92804 has AccuDraw for color art and tech drawings on the TS2068. All the features you'd expect to find in a first class drawing program. Supports full size printers through the Aerco printer I/F. Price \$19.95.

John Deering, 136 Neverbreak Drive, Hendersonville, TN 37075, is offering mods to PRO/FILE 2068 for use with the Rotronics Wafadrive System. The price is \$9.95.

John Riley, club librarian of the Capital Area Timex-Sinclair User's Group (CATS), has five volumes of TS2068 public domain software available for swapping with persons who donate new software to the library. For details, and a complete list of programs available, send a SASE to him at 1316 Farrara Dr., Odenton, MD 21113

Grey & Clifford Computer Products, P. O. Box 2186, Inglewood, CA 90305, (213) 759-7406, is offering software and hardware for communications on the Spectrum emulated TS2068. SPECTERM-64 is terminal emulation software which allows 64 column display without additional hardware. Price for SPECTERM-64 is \$30. When used with the new Z-SI/O hardware card, it allows use with any RS-232C modem or peripheral including Hayes compatibles. Price for the Z-SI/O is \$79.95. Write for more details.

Larry Kenny, Larken Electronics, has announced that he is working on a cart-bases DOS for the Larken 2068 disk drive. Features include Spectrum compatibility, built-in NMI "snapshot" save, usage of standard tokens, and disk compatibility with other systems. Also in the works are EPROMs for use with other popular disk systems. Contact LARKEN ELECTRONICS, RR#2, Navan, Ontario K4B 1H9, Canada, to be placed on the "inform-when-ready" list, or to suggest features that YOU would like to see.

Chia-Chi Chao, 73 Sullivan Drive, Moraga, CA 94556, has his latest catalog out. Listed are many great buys on hardware and software. Software titles include works by such companies as ABBA Software and JRC Software. Both game and utilities are available. Many of which are available on 5.25" Aerco disk. A complete catalog is available with SASE.

AN-TO Productions, 9009 West Elm St. #2, Phoenix, AZ 85037, is currently marketing QUICKKEY 2068 keyboard overlays. Overlays are available for the following programs: AccuDraw T2, Artworx v1.1, Tasword/Tasprint, Mscript (regular and V5), and Omnicalc II. There is also a keyfinder and a blank overlay for your own programs. \$6.95 each.

Fred Nachbaur, Silicon Mountain Computers, C-12, Mtn. Stn. Group Box, Nelson, BC V1L 5P1, Canada, has been hard at work under the mountain and has finally completed DUNGEON OF YMIR V3 for the ZX81/TS1000 with CMOS 8-16K RAM and a minimum of 16K of user RAM (64 K Rampaks will also work provided the 8-16K region is disabled and the CMOS RAM connected). The circuit published in SWN 4:1 is adequate, as is the Hunter NVM (modified as per SWN 4:2). Price \$24.95 U.S.

Bill Bell, 596 Cherrington Road, Westerville, OH 43081, (614) 882-3883, announces the availability of an Oliger Video Version of BBDOs for the TS1000 equipped with the Aerco FD-ZX Disk Interface and the JLO Video System. BBDOs is a fully automatic, BASIC transparent, disk operating system that locates itself in the 8-16K region of RAM. This BBDOs version enhances the operation of both the AERCO and JLO additions to the TS1000. It is fully compatible with all printer interfaces. \$29.95.

G. Russell Electronics, RD 1 Box 539, Centre Hall, PA 16828, (814) 364-1325, has NUCLEON (\$34.95) and TASKMASTER in stock. They are also offering some special sales on QL software. Spectrum ROMs for the TS2068 are currently available for only \$16.95. Call or write for details.

RMG Enterprises, 1419 1/2 7th Street, Oregon City, OR 97045, (503)655-7484, announces the publication of a new book, ARCHIVE MASTER. This new book delves into the inner workings of the QL program Archive and contains over 100 tested procedures, tips, and tricks to help you make Archive an even more powerful working tool. Published in loose leaf format, it comes in its own three ring binder. \$44.95

Weymil Corporation, Box 5904, Bellingham, WA 98227-5904, announces NOVA 1000, a multi-tasking program for the TS1000. Features include a real-time clock, automatic program line tracer, auto-repeat on all keys, and more. Price \$20.00. TRACER TS2068 is a debugging tool for BASIC programs. It allows you to see each line of your BASIC program as it is executed. Debugging has never been easier! Price \$15.00. SOUNDESIGN TS2068 is a sound development program for the TS2068. This program will allow you to manipulate the sound chip in ways you never thought possible. Once you have designed the sound you desire, a printout feature lets you obtain a hardcopy for use in other programs. Price \$ 15.00

The Cuyahoga Valley Software Works, 615 School Ave., Cuyahoga Falls, OH 44221, is currently marketing SPDOS for the TS2068 and the J L Oliger Disc Interface. SPDOS is available for 40 and 80 track DS drives, supports sequential files, and offers a fast, versatile operating system that requires less than 8K of RAM. Fully compatible with the Oliger memory resident DOS, it will also read and run programs written for the Millenia K Interface. Supplied on two disks. Price \$24.95 plus S&H. Write for details.

FORUM

Come to the Fest !

From the Desk of Paul Holmgren, Executive Chairman

The Second Annual Midwest Timex-Sinclair Computer Fest is a definite reality. We are committed to making it even more of a success than the show we put on last year in Cincinnati, Ohio at the Ramada Inn. Please accept this letter as a personal invitation for all the readers of SyncWare News to attend what will undoubtedly be the PREMIER event for the Vendors and Users of Timex, Sinclair, and QL computers in North America. Those who attended last year's show in Ohio know exactly what I mean.

This year's show will be held at the Holiday Inn-North, located just off I-465 on the northside of Indianapolis, Indiana. The mailing address of the Inn is 3850 DePauw Blvd., Indianapolis, IN 46268. The local phone number is (317)872-9790. The national toll-free number for reservations is

SyncWare

Thomas B. Woods
Jeffrey D. Moore
Fred A. Nachbaur
Thomas J. Bent

Publisher
Editor
Technical Director
Asst. Editor

All subscription information, billing, and ad artwork should be addressed to:

SyncWare News
PO Box 64
Jefferson, NH 03583
(603) 586-7734

Article submissions, Forum, Support listings, Classified ads, and really easy questions should be addressed to:

SyncWare News
603 S. Mill St.
Louisville, OH 44641
(216) 875-1257

Article submissions, hardware, and everything else, especially the tough questions, should be addressed to:

SyncWare News
C-12, Mtn. Sta. Group box
Nelson, BC V1L 5P1
(604) 354-3858

Back issues of SWN are available for \$3.50 each. Volume #1 (for the 1000) is available for \$16.95.

SWN is done entirely on TS computers. Submissions are preferred as word processed text files recorded on good quality tape. Memotext for the 1000 and Mscript for the 2068 are the preferred word processors. Documents submitted in Tasword Two and other word processors must be done in 52 column format. If you do not have a word processor, but you feel you have a good idea, then don't hesitate too work it up and submit it. Complete writers guidelines are available upon request from the Ohio Editorial office. SWN pays authors about \$40.00 per page for original articles.

1-800-HOLIDAY. We have made arrangements to obtain commercial room rates, which offer substantial discounts for those attending the Fest. There will be a flat rate of \$52.00 per night, but to obtain this rate you have to state that you are attending the Fest when you make your reservations. The flat rate will apply regardless of the number of guests staying in your room. The site is a Holidome-- a complete full service hotel facility with all the expected amenities: indoor heated pool, jacuzzi, separate men's and women's saunas, billiards, ping pong, a putting green, and wide screen TV. Along with all that, you'll be treated daily to complimentary fresh coffee and morning newspaper! For those flying in, the hotel offers bus, van and station wagon shuttle service to the airport.

Enough about the hotel, let me get on to the Fest itself!

We have been allocated approximately 6000 square feet of space not including the Hospitality Suite that will be available for you to take a break from it all. This is over twice the space we had for the Fest last year. We are planning two sets of lectures and seminars which will be running concurrently. We hope to be able to repeat the ones which we have had the highest demand for. There will be ample space for the numerous vendors, user groups, magazines, newsletters, and cottage industry vendors to show the Fest attendees just what they have to offer. There will also be an area set aside as a swap and/or flea market area. This will allow attendees to buy, sell, or trade used hardware and public domain or used (NOT PIRATED) software. This area will be available free of charge but will be rigorously policed and the rules strictly enforced.

Other motels available in the immediate area that are:

- 1) Dillon Inn (800)253-7305 or (317)875-7676
- 2) Dollar Inn (317)872-0500
- 3) Drury Inn (800)325-8300 or (317)876-9777
- 4) Red Roof Inn (800)848-7878 or (317)872-3030
- 5) Signature Inn (800)822-5252 or (317)875-5656
- 6) Embassy Suites (800)362-2779 or (317)872-7700

MEMBERSHIP RATES:

Individual: \$6.00 Family: \$9.00

DEALER ROOM RATES:

Hardware/Software Vendor.....\$65.00 per table
Software/Non-computer Vendor....\$25.00 per 1/2 table
Absentee Dealer, Vendor, Mags...\$25.00 per 1/2 table
User Groups.....\$15.00 per 1/2 table

For more information concerning the Fest, write the Fest Committee at:

The Midwest Timex-Sinclair Computer Fest
513 East Main Street
Peru, IN 46970

Thank you! See you all at the Fest!!!

2068 Wish List

Dear Editor, Several times I have read requests by the Editors of TS magazines for input from the readers about the hardware and software they'd like to read about. Since I know very little about electronics, I'm not even sure these items are feasible, It is just my wish list.

- 1) A real time clock for the 2068 to allow the machine to run programs or interfaced machinery at specific times.
- 2) A DC power supply for the 2068 mated with a 9 inch monochrome or RGB monitor and a microdrive.
- 3) The 32K NVM RAM PAK upgraded to 64K.
- 4) An expansion buss to allow several NVM's to be connected together and then to be bank switched.
- 5) Some way to network the TS2068 with other 2068's and TS1000's. Maybe using the TS1000 for home control or robotics purposes.
- 6) A TS User group in the Hattiesburg, MS area which is near Biloxi, Jackson, New Orleans, and Pascagoula.

Pete Kelly
Hattiesburg, MS

Pete-Thanks for your 'wish list'. All your ideas seem like good ones. Most great inventions that we take for granted today began as someone saying 'What if?'. Take that fellow Clive Sinclair for example....-ed.

ZXLR8 Woes

Dear Editor, I purchased and installed a Delphic 4K Programmer's Utility Eprom in my Byte-Back UM-64 memory pack. The EPROM resides in the 8-12K area of memory. Up to that time I was using the ZXLR8 fastload program. Now I can not use the ZXLR8 program without first disabling the EPROM. I have tried storing the ZXLR8 at other locations to no avail. I wonder if anyone would know of or have any suggestions as to how to solve this problem.

John Baney
Perrysville, OH

64 Column Mode

Dear Editor, As you know, the TS2068 has four display modes. In mode 2, the 64 column mode, there is 'garbage' printed to the screen every other column in the top half of the screen. Can you tell me how to clean up the screen using either hardware or software? Once the screen is cleared, how do I print to it?

John Pigg
Boise, Idaho

John- The 'garbage' you're seeing is the bit pattern of the data that was stored in RAM before you designated that area to be DISPLAY FILE 2. A good way to make use of the 64 column mode is to use a program like Wes Brzozowski's 2068 - 64 COLUMN BASIC that was the second place winner in the first SWN Programmer's Contest. The program was published in SWN Vol. 3 No. 1.-ed.

Key & Printer Problems

Dear Editor, I am having a couple problems using MSCRIPT with my GORILLA BANANA printer. First, I must be set up for two line feeds to get one to print. Second, when I do multiple page printing, the page spacing is erratic.

I would also like to find a vendor who can supply new keys for my 2068. The words and letters are worn off mine.

John Christensen
Oak Ridge, NJ

John- Your first problem, line feeds, is probably due to not having your printer set to the parameters of MSCRIPT or vice versa. Make sure that your printer performs an auto CR (carriage return) when MSCRIPT sends it a LF (line feed). Conversely, you may have MSCRIPT send both a CR and LF at the end of each line (at least with my version). Also, you need to make sure the LF and/or CR codes being sent by MSCRIPT are what your printer needs to perform the function you request.

*The second problem will probably be resolved when you correct the first, unless it is a mechanical problem with the printer paper feed mechanism. To fully advance an 11" long sheet of paper through the printer requires 66 LF's (ie., 6 lines/inch*11 inches). If MSCRIPT is providing anything other than 66 LF's/page, you will have erratic multi-page printing.*

I presently know of two methods for replacing worn or faded out keys on the 2068 keyboard. RMG Enterprises has replacement key tops which are just printing on adhesive backed, mylar sheets. You peel the legends from the sheet and stick them on to the key tops. RMG's address is listed in this month's SUPPORT column. The cost is around \$3.00. TS Connections, 3832 Watterson, Cincinnati, OH 45227 (513) 271-5575 sells replacement TS2068 top case halves. This would include new keys. The price is about \$10.00. Call Jack Roberts at TS Connections for details.-ed.

Dear Editor, I was excited about Bob Sworger's 2068 Label Maker in SWN 4:2. I tried running the program, then modifying the program for my printer. Unfortunately, I can not get the upper/lower scrolls to come out on print. I also get a lot of extra characters, and at times, NO line feeds (print over). I would like to use this program in future years, will someone please help?

My system consists of a TS2068, an A&J Mikrodrive and I/F, and a Legend 808 printer. I realize the Legend has limited capabilities, therefore the upper and lower scrolls do not have to be as indicated in Sworger's program. Simple "XXXXXXXXXX" will do.

Sergio Frost
San Diego, CA

Clubs

Dear Editor, I would appreciate it very much if you would send me a list of User Groups in my area.

T. G. Morley
Truth or Consequences, New Mexico

Mr. Morley- I currently have no listing for any user groups in the New Mexico area. I have sent you the names of two that might be close enough for you to attend their meetings. If you would decide to do something 'radical' like start your own, drop us a line and we'll announce it here.-ed.

Dear Editor, I am writing to ask if you would include the address of our user group somewhere in your publication. We would like to exchange newsletters with other user groups. Thanks.

Dallas Timex/Sinclair/Amstrad Users Group
c/o George Edmonds
P. O. Box 153421
Irving, TX 75015

Thanks

I'd like to thank the Greater Cleveland Sinclair Users Group for having me as their guest speaker February 6th at their East Side meeting. I had a most enjoyable time being with other TS people. Keep up the good work!-Jeff Moore, Editor

TECH MANUAL ERRORS

Dear Editor, I have discovered a number of errors in the schematic supplied with the TS2068 Technical Manual. I sent this list to Timex but they never replied. Perhaps this may be useful to your readers.

- 1) Resistor R2 in the U1-14 to Q1-E line should be in the Q1-B to Q1-E line.
- 2) ROSCS(NOT) is indicated on U3 to go to P1B-28 should be P1A-28.
- 3) VR-1 slider and lower end are grounded and the upper end goes to Q5-B & U3-33. The Draftsman untwisted the lines at the left end.
- 4) Put a connection 'dot' on the Q4-B line to the line between R55 and R54.
- 5) The added chip U21 is labeled 741500, it is a 74LS00.
- 6) U12-10 A7' is labeled AR7B. It should be A7RB.
- 7) Note 3 for D28128 should be D23128.
- 8) The buss lines around the Z-80 are incorrect. See the accompanying drawings.
- 9) The GND line drawn around Q5 is superfluous. This may be part of yet another error.
- 10) Which address lines are switched by U10 & U11 to make A1'-A6'? Eventually I'd like to ring these out.

Travis Bracken
Minneapolis, MN

Disk Expansion

Dear Editor, Can the Tandy or Purple 3-1/2" disk drive be used with a TS2068?

Jean Adams
Lummi Island, WA

Jean- Any disk drive that is 'Shugart compatible' will work with the TS2068 provided you have an interface between it and the disk drive. Disk interface boards are currently available from Aerco, Larkin, and Olinger, to name a few. Their addresses usually appear in the Support column or ads in this magazine. You must have a disk interface board to run a disk drive with any computer. The interface board must be specifically designed for type of computer its being used with.-ed.

Watch For...

Dear Editor, I am trying to locate a QL version of the program 'Solar Water Heater Design'. It was originally marketed for the TS1000 under the Timex-Sinclair label, listed as household, and as item #03-2017. The program was written by Tim Rossetti.

I had an original copy when I owned a TS1000. I would like to obtain either a version for the QL or a listing of the TS1000 version that I may convert for my QL.

Michael J. Pohl
306 Warren St.
Johnstown, PA 15905

Still using only 16K of TS1000 RAM? Now cheaply double or triple your memory and unleash your computer's true power.

Zebra Systems has purchased a large lot of new Memotech 16K and 32K RAMs directly from England so that we could offer them at drastically reduced prices.

Memotech RAMs are the highest quality and best-selling expansion RAMs for Timex/Sinclair computers. They have sleek looking black extruded-aluminum cases that conform to the shape of your TS1000 or ZX81. These 16K and 32K RAMs feature feed-through connectors and circuitry that allows you to use them along with the 16K RAMs you already have, so that you can achieve respectively, 32K or 48K total system RAM capacity.

Each RAM comes with an easy to understand instruction manual explaining how to use the RAM to best advantage, and a 15-day money-back guarantee from Zebra Systems. Order yours now, while the supply lasts.

Achieve 32K with
Memotech 16K
+ your Timex 16K

or

Achieve 48K with
Memotech 32K
+ your Timex 16K

CAT# R416 Memotech 16K RAM ... \$14.95

CAT# R432 Memotech 32K RAM ... \$24.95

CAT#S116 Sinclair 16K RAM, TS1016 equivalent... \$10.00

Include \$3.00 S&H. VISA/MC accepted. NY residents add sales tax.

ZEBRA SYSTEMS, INC.

78-06 JAMAICA AVE., WOODHAVEN, NY 11421, (718) 296-2385

Speak to Me

Dear Editor, Can anyone offer any tips or hints on how to use the word processor TASWORD in conjunction with the ZEBRA TALKER speech synthesizer?

Victor Philippi
Tupper Lake, NY

Wobble Revisited

We have had several letters from TS1000 owner's listing a whole myriad of woes which can all usually be traced to poor connections at the rear expansion connection. Based on personal experience, believe me when I say this connection has to be kept clean. It must also be kept free of movement when the computer is in use.

Cleaning can be done with a pencil eraser or with commercially available contact cleaner (one without a lubricant, oil will attract dirt). DO NOT USE metal cleaner such as that used on silverware or copper pots and pans! It will destroy the copper 'fingers' on the edge connector. G. Russell has some stuff that seems to be pretty good. It cured some nasty connection problems I was having on a TS2068.

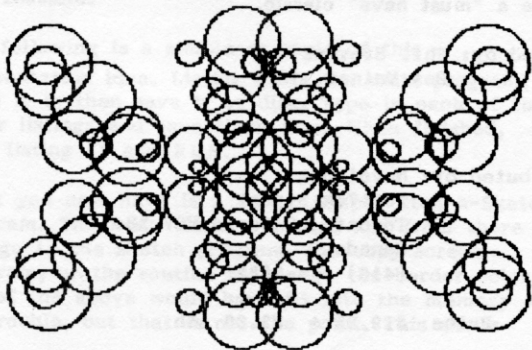
Anti-wobble devices can be anything as simple as rubber bands, to elaborate metal and/or plastic frames, to soldering the edge connectors together. Do whatever you feel necessary to prevent any one connection from being broke for even a millisecond. Just that short amount of time will cause a crash.

Dead End Space Bar

Dear Editor, I have a problem with my space bar not spacing if I hit it near the end instead of in the middle. Is this a common problem? Is there an accepted way of overcoming this?

James Holder
Huntsville, AL

James- Yes, this unfortunately is a common problem on the TS2068. A real cure involves some 'hacking' on the keyboard circuit board which is messy and not always successful. Some 2068 users put tape or otherwise mark off an inch at each end of the space bar. They then 'hit' the bar only between the marks. Another way to ensure good space entry, as well as all the other keys, is to turn on the keyboard 'click'. Do this by POKEing 23609,x; where x is a number from 0 to 255. The higher the number, the longer the 'beep' emitted for each key press. Start out with POKE 23609,64. Change this value as you see fit. Of course, this can be written into your own programs.-ed.



WRX16 Fix

Gregory C. Harder figured out why WRX16, as published, won't work with some 64K RAM packs. To quote from his letter,

"... There is no reason why the high-res routines won't work with 64K RAM. I've been using them with an Oliger 64K RAM board. You only have to make sure that the collapsed display file DUMY appears in high RAM; whether it's a phantom echo, as with 16K, or actual data in RAM doesn't matter. My Extended BASIC for this [under development -ed] simply loads it as data into high RAM before producing the display." -gch

This same technique has been used to make "Dungeon of Ymir V3" operable with 64K packs. Note, however, that you must still use a suitable NVM, and disable the 64K RAM in the 8-16K region, since WRX16 relies on getting the video data during refresh time.

I also realized why WRX16 will only work on the TS1500 if an external 16K pack is attached. The 1500's SCLD fully decodes the RAM space. Without the external 16K pack, the 1500's internal RAM is operating in the 16-32K region, and high memory contains nothing; i.e. no echo. However, when a 16K TS1016 is attached, the internal RAM is remapped to 32-48K, and the external pack runs in 16-32K instead. Since it is NOT fully decoded, it will of course generate the high-memory phantom, and WRX16 operates correctly. -fn

SALE!

 43% OFF!
WOW!

reg.
\$69.95

for ALL T/S
computers

IN/OUT PORTS

SALE! \$39.95

Experiment with computer sensing and control

The ideal port for sending signals from the outside world into your computer. Or use it to turn external devices on or off under computer control. 8 input lines, 8 outputs, plus 2 handshake strobes. Solderless female connectors. Now available at tremendous savings!

Thomas B. Woods
P.O. Box 64, Jefferson, NH 03583

***** SMART TEXT TS-2068 *****

*** A complete Administrative package: Word processor, Data base manager, Mail List, Full text editor, Automatic printer, Automatic formatter. Perfect letters or manuscripts in multiple copies. Mail List for personal form letters. Repeat print 5 page reports. Desk top publishing. Friendly, Menu Driven, no command keys to learn, Crash Proof.

*** Use with ALL Parallel Interfaces, Printers. Second program included for use with OS-64 Cart (64 Col Screen).

*** Disk Versions for Aerco FD and Oliger V2-12.

**** Satisfaction Guarantee. \$39.95 Check or M/O, Gulf Micro, 1317 Stratford Ave, Panama City, FL 32404. Specify type interface, for Cassette or Disk Drive, Inquiries welcome.

TIMACHINE Review

TIMACHINE: A floating-point BASIC compiler for the TS-2068 and Sinclair Spectrum. (C)1986
Written by: Cameron Hayne

From its animated title screen to its ability to locate compiled code anywhere in memory, TIMACHINE is just what every BASIC programmer who lacks extensive assembly language fluency has been dreaming of. You should see the wonderful things it does to graphics programs written in BASIC.

Compiling any BASIC program is as easy as adding a REM line at the beginning of the program, or portion of the program, you want to compile. Place another at the end, then key in *C. TIMACHINE performs its magic, pausing twice and awaiting any keystroke to continue. The first pass reports the relative addresses of the entry points to the code. The next displays the execution addresses in decimal and hex. At the completion of the run, the number of bytes used by the compiled code, bytes used for M/C variables, bytes occupied by the BASIC program before compilation and the commands necessary to save and load the compiled code are also displayed. All this occurs about as fast as you can press the keys. Your BASIC program remains intact unless it was too long to fit in memory, in which case you are asked for permission to overwrite it.

Input/Output commands are not included in the compilation, but passing from code to BASIC and back is easy. Simply insert the keyword STOP in your BASIC program then use the appropriate USR command to return to code. You are not prohibited from using any BASIC commands.

There are only minor restrictions in writing BASIC statements. All should be easy to program around. TIMACHINE supports both string and numeric arrays up to two dimensions. It easily handles the numeric VAL (expression) statements so commonly used in BASIC programs.

TIMACHINE is under 11K in length and loads quickly. Its small size compiles programs to 30K for the Spectrum and 27K for the TS-2068 without tape swapping. The program also contains numerous compiler directives. It even has a function to identify your variable types so that you may use the appropriate directives to produce the most efficient code.

TIMACHINE comes with excellent documentation. Included is a tutorial and example files. It is not copy protected, so it is a breeze to install on microdrives or a disk system. It also provides a backup copy feature immediately on loading.

Anyone who programs in BASIC or keys in BASIC listings from their favorite newsletters needs TIMACHINE. Even skilled assembly language programmers could benefit from the time they would save by writing in BASIC and letting TIMACHINE supply the speed of execution that only machine code provides. Among the many utilities written for the Sinclair computers, TIMACHINE is so well thought out and so versatile, that it is surely destined to become a "must have" classic.

reviewed by: Phil Stevens
Des Moines, IA

Distributed by: Novelsoft
106 7th Street
Toronto, Ontario M8V 3B4
Canada
(416) 259-8682

Price: \$19.95 + \$3.00 s&h

PROGRAM SEGMENTATION

BASIC TOPICS:

Program Segmentation with the Aerco FD68

by Bret Lanius

(EDITOR'S NOTE: While Mr. Lanius uses the Aerco FD-68 Disk System and MIKRO drives as examples in this article, the principles presented here in apply to any high speed mass storage device. Consider this, with the extra memory available on the Aerco FD-68 board (up to 256K on a modified unit), could a RAMDISK system also be feasible?-ed.)

We Timexers have grown accustomed to working within the memory limitations of our little machines. This has probably been out of a lack of knowledge and/or lack of mass storage devices that were fast enough to allow us to segment our programs. Segmenting a program simply means that only part of the program is actually in memory at any given time. This is how systems like CPM (RPM) can have programs that do much more than we Timexers would normally think possible in 64K. With Aerco's ZDOS operating system, there is no reason we cannot do this with our current systems.

Consider programs like ARTWORX or TECHDRAW. Both are very good programs, each with it's own set of features. Both of these is also limited due to the memory limitations of our computers. With segmentation, one could have numerous features incorporating the best of both. For example, Artworx has a UDG generator built in. You could have the program load in just that portion of program, then do the UDGs. When done, leave the UDG information in a safe location in memory, or save it to disk, then load back in the main program. While this would indeed slow down the program, you would gain the ability to have more options in a program. By limiting the segmentation to the less often used portions of a program, while keeping the main functions in memory, you would rarely see the speed difference.

This idea extends to other mass storage devices such as MIKRO DRIVES. For example, Tiny Board was originally a BBS written in the confines of the 64K memory on the Timex. The message base was limited to 20 messages. There are now Mikro Drive versions that make use of this segmentation idea and have message space limited only by the disk space available. Added features, like on-line adventures that would have otherwise been impossible to do in 64K, are now feasible.

The following is a simple example of this segmentation idea. Listing #1 is the main program. Type it in then save it to disk. Type in each of the other listings and save them also. When finished, load listing #1 and RUN.

What you now have is a VERY simple Etch-a-Sketch program. The main menu is loaded and from there you can go to the sketch program, the copy screen program, or the routine to change the border color. All of the above would have fit into the memory with no trouble, but that is not the point. This shows

how the work in progress can be saved to disk then recalled by another program.

While this may not be a very useful programming example, I hope that I may have planted a seed in the minds of those who dabble in programming with the Aerco FD-68 system. If anyone comes up with applications for this type of programming please let the rest of us know.

```
10 CLS : PRINT "      MENU"
20 PRINT "1) Etch a Sketch"
30 COPY "3) Border color"
40 IF INKEY$="" THEN GO TO 30
50 IF INKEY$="1" THEN CAT "Ske
tch.bas"
60 IF INKEY$="2" THEN CAT "cop
y.bas"
70 IF INKEY$="3" THEN CAT "Bor
der.bas"
80 GO TO 10
9999 MOVE "MENU.BAS",10
```

Listing #1 - MENU

```
10 REM sketch
12 CLS :
15 PRINT #1;"s=save q=quit l=l
oad"
17 PRINT #0;"Move with S,6,7,8"
20 LET l=0
30 LET c=0
35 PLOT c,l
40 IF INKEY$="" THEN GO TO 40
50 IF INKEY$="5" AND c>0 THEN
LET c=c-1
60 IF INKEY$="8" AND c<255 THE
N LET c=c+1
70 IF INKEY$="6" AND l>0 THEN
LET l=l-1
80 IF INKEY$="7" AND l<175 THE
N LET l=l+1
90 IF INKEY$="s" THEN MOVE "sc
reen.scr",: CAT "menu.bas"
100 IF INKEY$="l" THEN CAT "scr
een.scr",: GO TO 15
110 IF INKEY$="q" THEN CAT "men
u.bas"
9999 GO TO 35
9999 MOVE "sketch.bas",10
```

Listing #2 - SKETCH

```
10 CLS : PRINT AT 0,0;"Press 8
or 9 to quit"
20 INPUT "Border Color":a
30 IF a=8 OR a=9 THEN CAT "men
u.bas"
40 BORDER a
50 GO TO 10
9999 MOVE "border.bas",10
```

Listing #3 - BORDER

```
10 CLS : PRINT "Press a key wh
en ready to copy"
20 IF INKEY$="" THEN GO TO 20
30 CAT "screen.scr",: COPY
40 CAT "menu.bas"
9999 MOVE "copy.bas",10
```

Listing #4 - COPY

ABC-123 PART 2

The ABC Game

by Fred Nachbaur

In this installment, we'll finish off the "ABC/123" program package for the ZX81/TS1000, started in the last issue. This time we'll add an addition trainer (menu options 4-6), and a machine-coded arcade game (option 7).

The Machine Code

Start by entering the rest of the machine-code into the 0 REM statement at the beginning of the program. This time, we'll add almost 3K, so take your time and (preferably) save to tape after every screen. As before, use the hex-string loader (or a more sophisticated one of your own devise) to enter the eight screens of code (Table 2). Be sure to check the checksum at the end of each screen, and correct any discrepancies before proceeding to the next screen.

This sub-program was actually written in BASIC, and was then compiled into machine-code using the Bob Berch Compiler (still available from G. Russell Electronics). Since not all of the run-time package is used, and the program and variables are relocated from the default values, duplicating my compilation would be a little tricky. For this reason, we're publishing the object code instead of making you hassle with re-compiling the original source code.

After you finish entering the hex code, save the program to tape. You can then test the game by entering SLOW mode, then entering LET L=USR 18760.

The Basic

Now you've just got a little work left, that is to enter the BASIC lines of Listing 3. Since all of the key subroutines were already installed last time, adding the driver for the Alpha Space Bridge game (starting at line 4000) and the Addition drills BASIC (starting at line 2000) won't take you very long.

Finally, you may delete the hex loader/editor (lines 7000-7370) to save a little space and loading time. Save the finished program to tape; you're done!

Simple Addition

Options 4, 5 and 6 will now be active, and form the second sub-program within ABC/123. Again, several levels within the options provide flexibility and a gradual increase in difficulty. This sub-program displays a pair of large digits (from 0 to 9), and prompts for their sum with a question mark. The format is vertical, with a line below the numbers, just as the student will encounter such problems in school.

Keyboard response is, unfortunately, a little slow; it takes a deliberative keypress to "register." To

help counteract this, the screen flashes when the computer recognizes a valid keypress. Encourage the student to hold the key until the screen flashes.

If a single-digit answer is required (e.g. $5+3=8$), then any wrong keypress will result in the infamous "sad face." If a two digit answer is required (e.g. $8+4=12$), the "1" is recognized as the only valid first keypress, and the program waits for the second digit. (The total of two single-digit numbers will of course never exceed 18.)

You might notice that the number zero is included in the number line, and in the problems. This reflects my disagreement with other teaching methods (including Sesame Street), which studiously ignore the zero. Zero is an important number, and the earlier that the student recognizes its significance, the less trouble he will have later when dealing with multiple digits, and (much later) with negative numbers.

The overall format is much like the ABC portion of the program, as is the training emphasis. As before, a score of seven or more robots advances to the next level, whereas less will cause that level to be redone.

4: Addition Drill

This option is used to drill sums involving specific numbers. When you select this, you are asked which number to start drilling. If you select "1," for example, it will drill $1+0$, $1+1$, $1+2$, ... $1+9$. The second number is in random sequence, to prevent "cheating" by simply scanning across the top row of keys. Each number from 0-9 appears only once, for a total of 10 problems per "set."

After you enter a "Drill which?" number, you are asked "Last Number to Drill?". To continue the example, if you enter "1" for "Drill which number?" and "3" for "Last number?", then the drill will advance to "2" when "1" has been adequately completed, and then to "3" before ending off.

To the left of each number, a row of inverse stars is shown, corresponding to that number. If the pupil gets stuck, he can simply count the stars to get the correct answer. This is to help convey the concept of addition as the totalling of two sets of identical items.

Subtraction is not catered to in this program. However, if you study how the addition portion works, you shouldn't have much difficulty implementing a similar subtraction drill.

5: Scramble1

This is similar to option 4, except that all numbers 0-9 are "fair game," and are used in ten additions problems. The problems are randomly generated, but each number appears twice; once in the top position, once in the lower position.

TABLE 2: HEX-CODE FOR ABC/123, PART 2

START = 18760

21010022F453211A0022F65321010022
F85321604922FA531125002AF45319E5
2AF453CDF747C053CDDF48F453CDA4A
2AFC5322FC53CD0E492AFE53E52AFC53
D1CD2347CD8248120013702AFC53CDB9
48CD8248078084702AFC53CDB948CD82
48800380702AFC53CDB948CD82488180
82702AFC53CDB948CD8248070084703E
76D7CD0E492AFE53E521000D1CD2347
21010022F45321050022F65321010022
F85321F04922FA532AFC53CDB948CD82
4800000070CDDF48F453CD0E49210B00
E52A0054D1CD2347CD824800838A8300
000000000000000000000000000000
000000000000000000000000000000
000000000000000000000000000000
54CD2847CD8248808200703E76D7CD88
4A21010022F4531105002A0454CD4247
EB211E001922F65321010022F853216C
4A22FA53CDDF48F453CDB24A1189002A
0654CDB547E5211700D1CD0B49CD0E49

CHECKSUM = 30585

START = 19080

ED5BFC532A06541911A600CDB547E521
1700D1CD0B491189002A0654CDB547E5
219700D1CD0B49CD0E49ED5BFC532A06
541911A600CDB547E5210000D1CD0B49
1189002A0654CDB547E5211700D1CD0B
49CD0E491102002AF453CD4647CDA4A8
884A1101002A085419220854C3B24ACD
0E49CD2A0A210500E5210000D1CD2347
CD8248888887021260022F453213F00
22F65321010022F85321274B22FA532A
F453CD2847CDDF48F453CD8248888888
888870211E00CDB948CD82488888703E
76D721010022F45321030022F6532101
0022F85321624B22FA53CD8248887021
1F00CDB948CD824888703E76D7CDDF48
F453210000E5211E00D1CD2347CD8248
1770211D00CDB948CD8248878A047021
1E00CDB948CD82488A70211D00CDB948
CD824885970570211D00CDB948CD8248
979797703E76D721010022F453210A00

CHECKSUM = 31264

START = 19400

22F65321010022F85321D74B22FA5321
0000E5210800D1CD2347CD82489780B8
B5A6A8AA80A7B7AEA9ACAA809770CD8E
0B211500E5210300D1CD2347CD82481D
14272A26322A371A0021DA24DF32343B
2A703E76D7210100220A54210A00220C
54210100220E5421354C221054CD1B48
CD9D47111A00CD4647EB2101001922FC
532AFC53CDBF47C053221254CD1B48CD
9D47111A00CD4647EB21010019220054
2A0054CDBF47C053E52AFC53CDF747C0
532A1254E52A0054CDF747C053CDDF48
0A54210000E5210800D1CD2347CD8248
000000000000000000000000000000
703E76D7CDDF48F453211500E5210300
D1CD2347CD8248312A3B2A310F001D00
102A26383E1100162500102D26372911
703E76D7CD2F49220454111D002A0454
CDD147E51125002A0454CDD047EBE1CD
F047CDA448DC4111C002A0454CDB547

CHECKSUM = 27671

START = 19720

EB213200CD4A47EB212602CDB5472204
5421020022F453211A0022F653210200
22F85321314D22FA53211300E5110200
2AF45319D1CD23472AF453CDBF47C053
11800019CD28473E76D7CDDF48F45321
010022F45321190022F65321020022F8
53216F4D22FA53211500E51101002AF4
5319D1CD2347CD824800702AF453CDBF
47C05311800019CD28473E76D7CDDF48
F45321100022F45321150022F6532101
0022F85321B24D22FA532AF453E52100
00D1CD2347CD82488580703E76D7CDDF
48F45321000022FC53210D0022005421
0000221454210C002216542101002218
54210100220854219700220254210000
22F853210C40CD1249E5210D40CD1249
EB210001CD4A47EBE11911A800192206
54CD8149CD054A112400CD2F49CDB947
E5111A002A0054CDD147EBE1CDE547CD
AD48464E1101002A0054192200541121

CHECKSUM = 27600

START = 20040

00CD2F49CDB947E51101002A0054CDD0
47EBE1CDE547CDAD486F4E1101002A00
54CDB547220054111D00CD2F49CDB947
CDA448C14E1101002A085419220854ED
5B04542A0854CDD147CDA448BE4E2100
002208541101002A1654CDB547221654
210200E52A1654D1CD95482A1654CDB
47CDA4487F51C31C4E2A1454CDA44883
4F210C0022F45321150022F653210100
22F85321E14E22FA53CDDC4A2AF453E5
1102002A005419D1CD2347CD82488870
3E76D7CDDF48F4532101002214542A00
54CDBF47C053221A54210000E52A0054
CDF747C053ED5B1A5421800019220254
21140022F453210C0022F65321FFFF22
F85321404F22FA53CDDC4A2AF453E511
02002A005419D1CD2347CD824888703E
1102002A005419CDB948CD824800703E
76D7CDDF48F4532A1A54CDB47CDA448
9350210100221454C31C4E1125002A00

CHECKSUM = 29408

START = 20360

5419ED5B1A54CDB947221C54210B0022
F453ED5B1C54210600CDB54722F65321
FFFF22F85321B34F22FA53CDDC4A2AF4
53E51102002A005419D1CD23472A0254
CD28471102002A005419CDB948CD8248
88703E76D7CDDF48F4532A1C54CDAD48
F04F21880022025421060022F453210B
0022F65321010022F85321085022FA53
CDDC4A2AF453E51102002A005419D1CD
2347CD824800701102002A005419CDB9
482A0254CD28473E76D7CDDF48F4532A
1C54CDB47CDA4481C4ECD49110100
2AFC531922FC53CD8149ED5BFC532A06
5419CD1249118000CDD147E5ED5BFC53
2A065419110100CDB547CD1249118000
CDD147EBE1CDE547CDA448A25011A00
2AFC53CDB947CDA44891522100002214
54219700220254C31C4ECD054A211100
221C541101002A005419ED5BFC53CDB5
47CD9D47110400CDD147CDAD48CD5021

CHECKSUM = 31181

START = 20680

0700221C5421000022F4532A1C5422F6
5321010022F85321E55022FA532AF453
22FE531101002AF53CDB547E52AFE53
CDBC47EBE119E52AFC53D1CD2347CD82
48000000703E76D7CD81491102002AF4
53CD4647CDAD4838512AFE53E5110100
2AFC5319D1CD2347CD824817703E76D7
CDDF48F453210100E5210800D1CD2347
CD824827372E292C2A002E3800273734
302A3370210300E5210900D1CD2347CD
8248B7B4A7B4B980ABAAB1B180B4ABAB
703E76D7C3A051CD054A210700E5210A
00D1CD2347CD8248B4BAB980B4AB80B5
B4BCAAB7703E76D721010022F4532114
0022F65321010022F85321B85122FA53
1101002AF5319E52AFC53D1CD2347CD
8248808080703E76D7210100220A5421
1400220C54210100220E5421E9512210
54CDDF480A541101002AF5319E52AFC
53D1CD2347CD8248898089703E76D721

CHECKSUM = 32241

START = 21000

0100220A54211400220C54210100220E
54211F52221054CDDF480A541101002A
FE5319E52AFC53D1CD2347CD82480780
84703E76D7210100220A54211400220C
54210100220E54215552221054CDDF48
0A54CDDF48F4532AF53E52AFC53D1CD
2347CD8248001800702AFC53CDB948CD
8248130013702AFC53CDB948CD824806
0717703E76D7CD0E49CD054A2A165422
0A54210B00220C54210100220E5421AC
5222105421010022F45321050022F653
21010022F85321C45222FA53210000E5
211A00D1CD2347CD8248131712149714
70210000E5211A00D1CD2347CD824812
151316171670210000E5211A00D1CD23
47CD8248120013001700703E76D7CDDF
48F453210200E52A0A54D1CD9848CDDF
480A5421010000E5210900D1CD2347CD82
48B7B4A7B4B980B7AAABBAAB1AAA970
210300E5210600D1CD2347CD8248322E

CHECKSUM = 27525

START = 21320

38382E343300262828343235312E382D
2A29703E76D7ED5B0454218A02CDB547
E5ED5B1654210A00CD4A47E5ED5B0854
210A00CD4A47ED5B0454CD4247EBE1CD
B547110A00CD4247EB21030019EBE1CD
4A4722FC53211400E5210A00D1CD2347
CD8248382834372A001400702AFC53CD
42483E76D7CD0E492F003C002B003F00
3D002A0030003B0027002E0034000000
2D003E00000036003A00320037002800
330039003500260029002C0015001400
0100B85102001100130088006400156F
41001500140001005552280001000C00
010038001100007600010C00EC251C1C
CHECKSUM = 13801

Successful completion of ADDITION SCRAMBLE I (seven or more robots) automatically advances to ADDITION SCRAMBLE II.

6: Scramble2

This is like option 5, except that the "stars" that allow counting the total, are not displayed. Mastery of this level makes the pupil ready to go on to two-digit addition, as available on many existing "maths drills" programs.

7: Alpha Space Bridge

The last sub-program is a game called "Alpha-Space Bridge." Robbie the Robot starts out at the left end of his "space bridge," and has to cross it to his power station, so that he can refuel. The bridge hasn't been completed yet, which is where you (or rather, your pupil) comes in. Nothing but the merest framework exists (shown as the alphabetic letters A-Z). At the bottom of the screen are the "bricks" that have to be installed in the bridge (the inverse letters A-Z). The problem is that they're all mixed up, so you have to flit back and forth in your flying saucer, picking up the right bricks with your "beamer" and installing them in the proper place on the bridge.

Only three keys are used to play the game: the "5" and "8" keys (with the arrows on them) to fly back and forth, and the "1" key to activate the "beamer." The beamer will pick up whichever brick your saucer is positioned over, and bring it into the "cargo hold." You then have to install it by flying directly below where it should go, and again energizing the beamer. If you try to install the brick in the wrong place, it will simply bounce back down.

Robbie is something of a simpleton, and can't tell if the part of the bridge he's about to step on has been completed. However, he senses whenever you install a brick, and automatically takes a step after each installation. The bridge will break, and Robbie will take a dive (ending the game) if he is standing on more than one unfilled space. He therefore trusts your pupil to have bricks installed where he is about to step.

Meanwhile, Robbie's energy level (shown as a fuel gauge in the lower left corner) ticks inexorably away. If he runs out of juice before making it to the power station, his eyes start to flash ominously before he self-destructs. End of game, try again. If he makes it across before his fuel runs out, he makes contact with the power station and his gauge fills up again, while his antennae flash happily.

The final score depends on how much fuel he had remaining on reaching his destination, and on the skill level selected. The skill level is specified at the outset of each game; higher numbers are more difficult. The skill level selects the overall pace of the game, and also (more importantly) the rate at which Robbie's energy depletes. Though the scoring is weighted to reward higher skill levels, it has been carefully balanced so that a younger child can successfully compete with an older friend or sibling who selects a higher skill level. At the end of each game, the highest score thus far is displayed.

```

2000 REM ADDITION
2010 LET Y$="U"
2020 LET D=0
2030 FOR N=N1 TO 10
2040 LET E=VAL S$(N)
2050 LET F=VAL T$(N)
2060 LET T=E+F
2070 PRINT AT 0,0;N
2080 IF F$<"6" THEN PRINT AT N4,
0;X$( TO N2*E)
2090 LET M$=S$(N)
2100 LET R=N3
2110 LET C=25
2120 GOSUB N2
2130 LET M$="+ "+T$(N)
2140 LET R=8
2150 LET C=17
2160 GOSUB N2
2170 IF F$<"6" THEN PRINT AT 9,0
;X$( TO N2*F)
2180 PRINT AT 13,17; "
";AT 15,21;
2190 IF USR BT THEN REM <>0 ?
2200 GOSUB 20
2210 RAND USR REV
2220 LET M$=A$
2230 LET R=16
2240 LET C=25
2250 GOSUB N2
2260 RAND USR REV
2270 IF T<=9 THEN GOTO 2360
2280 IF A$<"1" THEN GOTO 2370
2290 GOSUB 20
2310 RAND USR REV
2320 LET M$=M$+A$
2330 LET C=21
2340 GOSUB N2
2350 RAND USR REV
2360 IF VAL M$=T THEN GOTO 2420
2370 GOSUB 600
2380 LET LEN=30
2390 GOSUB PA
2400 GOSUB 700
2410 GOTO 2180
2420 GOSUB 500
2430 LET LEN=25
2440 GOSUB PA
2450 GOSUB 150
2460 GOSUB 850
2470 NEXT N
2475 GOSUB 800
2480 GOSUB 300
2490 GOSUB 920
2500 GOSUB 150
2510 IF SC<8 THEN GOTO 1330
2520 IF F$<"4" THEN GOTO 2560
2530 IF ST=FI THEN GOTO 8800
2540 LET ST=ST+N1
2550 GOTO 1440
2560 IF F$="6" THEN GOTO 8800
2570 LET F$="6"
2580 GOTO 1440
4000 REM ALPHA-SPACE BRIDGE
4005 SLOW
4010 LET SC=USR 18760
4015 IF SC<26 THEN GOTO 4070
4020 IF NOT HIGH THEN LET HI=SC
4030 IF SC=HIGH THEN GOTO 4060
4040 LET HIGH=SC
4050 PRINT AT 16,0;"YOU GOT THE
HIGH SCORE, ";N$
4060 PRINT AT 15,8;"HIGH SCORE="
;HIGH
4070 LET LEN=100
4080 GOSUB PA
4090 GOSUB 800
4100 PRINT AT 3,8;
4110 IF USR BT THEN REM <>0PLAY
4120 PRINT AT 8,6;
4130 IF USR BT THEN REM <>0AGAIN
?
4140 IF INKEY$="" THEN GOTO 4140
4150 IF INKEY$="Y" THEN GOTO 400
0
4160 GOSUB 150

```

This game is especially suited for children who have outgrown the earlier ABC drills. Tell them the basic scenario, but let them discover the "quirks" themselves. It won't be long before your pupil realizes that he only needs to install every OTHER brick at first, filling in the rest at his leisure. As he gets more proficient, he'll learn to plan in advance which bricks he picks up, minimizing useless flying around. It also won't take him long to realize that, once a brick is picked up, it cannot be put back and must be installed in the proper location. He'll be amused with the graphics,

especially when Robbie falls to his demise because of "contractor error."

Mods

If you're adept at BASIC programming, you'll have little trouble adapting the program however you see fit. The program starts with the initialization and Parent/Teacher section at line 9000. From there it goes to the appropriate sub-program; 1000 is the start of the character recognition program, 2000 is the addition program, and 4000 is the Alpha-Space Bridge game. A system of flags is used to keep track of the various sub-levels. Variables are used to store the various USR addresses for the screen utilities, and to store the commonly used numbers 1-5 to economize memory. The "big letters" are done, as mentioned earlier, using Greg Harder's "SW-LO-RES," published in SWN 3:1. It was modified only slightly to refer to the self-contained character patterns instead of the ROM patterns.

Even if you're not into BASIC hacking, there are a few simple changes you can make to modify the program. To change the scoring, edit line 320 and change the "10" to another number. This number divided by 5 gives the number of seconds (average) corresponding to one robot in the score/reward at the end of the drills. So, to make the scoring more generous, change this to, say, 20. This will especially be necessary if you are using the Oliger video upgrade. To make the scoring more difficult, decrease the number to perhaps 5 (1 robot = 1 second).

If you don't like my happy/sad faces, change them by modifying the PRINT statements in subroutines 500 and 600. If you make them wider than 7 spaces, you'll also have to change the number of spaces printed by the "erase face" subroutine at line 730.

To change the scoring criterion for advancing to the next level (currently set at 7), change the number in lines 1320 (character trainer) and line 2510 (addition trainer).

If you don't want the program to print the problem number in the upper left corner, delete lines 1040 (character trainer) and/or line 2510 (addition trainer).

To save the program to another medium (fast-load tape, disc, etc.), change the SAVE command in line 9990 appropriately.

There is about 3K of unused space in 16K, so you have plenty of room to add your own touches, perhaps even other sub-programs.

I found it very rewarding to see my little friends using and enjoying this program; their smiles made it worth the effort. I'm sure that your little friends will likewise reward you for your efforts in typing it in, and perhaps developing your own applications for the SWN Program Contest.

CHROMA SOFT TS1000 Review

Chroma Soft

G. Russell
R D 1 Box 539
Centre Hall, PA 16828

Price: \$14.95

Color from a black and white TV and an unmodified TS 1000, impossible! Then I booted up Bill Russell's Chroma Soft. It really works! Chroma Soft does produce color graphics and text on an unmodified ZX81/TS1000 with 16K ram.

The color palette consists of nine colors: 2 reds, 2 blues, yellow, green, cyan, black, and white. The amazing part is how and where the color is produced. The colors are created in your mind. This program tricks your mind into "seeing" color by flashing bit patterns at different rates. Because of this subjective coloring, some of the colors are easier to see than others. The user's mental and physical state have a lot to do with how well and how many of the colors he or she sees.

Chroma Soft comes with a program tape, Chroma Soft filter, and an 11 page instruction manual. The filter is attached to your black and white TV. This must be done to view the coloring effect. The manual is required reading to get the most out of this program.

It takes about 6 min. for the tape to load. When finished loading the user is presented with a full color demo screen. Pressing "A" gets you to the main menu where you are prompted to choose a color.

Some planning is required before drawing your color screen. Colors are drawn one screen at a time. The different colors are not allowed to overlap. There are two methods for drawing, coded string and direct. The direct method is the simplest to use. The cursor is moved around the screen and points are plotted or unplotted by pressing a 1 or a 0. The coded string method takes a little practice but is the most versatile. This method allows the use of the whole character set which gives greater flexibility to your drawings. The on screen help pages are useful, but the user is always referred to the manual for further information.

The instructions are clear and well written. Many hints on use and modification to achieve better or different results are presented for the user to fine tune or experiment with the colors. The methods used in the generation of the colors along with an annotated machine code listing and program entry points are explained in detail.

In conclusion, I found Chroma Soft a very interesting and unique program. Dig out those TS1000/ZX81's and give this one a try. You will be pleasantly surprised as well as intrigued by it.

Reviewed by Kenneth E. Majors

2068 PIANO

Basil Wentworth

Here's a program that lets you play the TS 2068 like a piano, by using the BEEP function. Or maybe I should say like a xylophone, since the instrument will not sustain notes, but rather gives the impression of sustaining by repeating a note rapidly.

First, let's have a look at what the BEEP does. You will remember that BEEP needs an argument of two numbers, the first specifying the duration of the note, and the second defining its frequency (pitch). If 0 is used as a pitch signal, middle "C" is generated. The pitch rises a half tone for each time that number is increased by 1; it goes down a half tone for each time the number is decreased by 1. (Fractional increases or decreases are permitted if your ear will resolve intervals smaller than a half tone).

NOTE: in the tempered scale to which pianos (and TS 2068's) are tuned, a half tone (also known as a semi-tone) represents a frequency ratio equal to the twelfth root of two.

I should mention here that there is an error on page 66 of the 2068 manual. Numbers 4 and 5 should be replaced by 3 and 4, respectively, and "FA and SOL" should read "MI and FA".

You can program a major scale into your computer with the following mini-program:

```
10 LET F=0
20 FOR G=1 TO 15
30 BEEP 1,F
40 IF G=3 OR G=7 THEN LET F=F-1
50 IF G=10 OR G=14 THEN LET F=F-1
60 LET F=F+2
70 NEXT G
```

A little experimenting with the length of the note showed that .05 was about the shortest I could make the constant in line 30 and still have a pitch defined well enough to suit my ear. If the duration of the BEEP is too short, it degenerates into a click. If the note is too long, then operation will be sluggish. Fool around with it yourself, and see what value suits you best. If you want, you can make it flexible, to be set initially each time you RUN the program. Or you could even make the length of the tone depend on its frequency, by setting up a second array to store a tone length for each note.

As you can see from the listing, the program itself is very simple. The "working part" of the program is contained in lines 100 to 140--the rest of the program is used only once, to set up a matrix of notes to correspond to the computer's keys.

The line numbers used in lines 634 to 795 are not as capricious as they might look. If you will compare

them with lines 520 and 550, you will see that their line numbers are related to the positions in array A, so that A(34) becomes the first number in line 634, A(195) becomes the first number in line 795, and so on.

When the program is first RUN, the FOR-NEXT loops in lines 520 to 570 (and the single-step "loop" of line 580) assign the appropriate values to array A, and then let you know when the "piano" is ready to play. You will find then that the lower-case values of the bottom two rows of keys give you an octave of notes (C to B), which is continued for a little over an octave on the top two rows. The lower of the two rows acts as the piano's white keys in each case, while the upper of each pair of rows serves as the black keys.

It will help you to keep them straight if you identify the black keys in some way, as by sticking a square of masking tape on keys S, D, G, H, J, 2, 3, 5, 6, 7, 9, and 0. But don't use anything (like fingernail polish, for instance) that could damage the surfaces of the keys, or would be hard to remove.

To extend the range of the instrument, the symbol shift key makes all notes sound an octave higher than their normal pitch, while use of the capital shift lowers by one octave the pitch of all notes played on the lower rank of keys.

```
10 GO TO 500
100 IF INKEY$="" THEN GO TO 100
110 LET B=CODE INKEY$
120 IF B<1 OR B>205 THEN GO TO
100
130 BEEP .05,A(B)
140 GO TO 100
500 DIM A(205)
510 RESTORE
520 FOR F=34 TO 121
530 READ N: LET A(F)=N
540 NEXT F
550 FOR F=195 TO 205
560 READ N: LET A(F)=N
570 NEXT F
580 LET A(172)=36
585 CLS: PRINT "READY"
590 GO TO 100
634 DATA 40,27,0,30,32,34
640 DATA 0,37,19,0,21,22,23,17,
27,0
650 DATA 13,15,0,18,20,22,0,25,
12,38
660 DATA 29,0,31,16,25,0,-5,-8,
-9,0
670 DATA 0,-6,-4,0,-2,0,0,-1,-3
,0
680 DATA 0,0,0,-11,0,0,-7,0,-10
,0
690 DATA -12,0,0,0,20,39,14,0,7
,4
700 DATA 3,16,0,6,8,24,10,0,0,1
1
710 DATA 9,26,28,12,17,1,19,23,
5,14
720 DATA 2,21
795 DATA 13,0,35,33,24,28,26,0,
18,0,15
```

Listing 1 - The 2068 Piano

ANOTHER VIDEO PRIMER

by Fred Nachbaur

Our mailbag lately contains a lot of questions about various aspects of video. There appears to be some confusion about RGB, composite, and monochrome monitors, how these differ from television, what is meant by "resolution," and so on. In SWN Vol. 1 ("Journal #1") we ran an in-depth article on monochrome ("black-and-white") video, and I still recommend this Video Primer for anyone wishing to learn more about video systems. However, this ZX81-oriented publication is not a prerequisite to this article. Since it was limited to monochrome, the previous article sheds little light on the various aspects of color video. I also recommend that you look over the back-issues of SWN Vol. 2, 3 and 4, since several specific video-related questions were answered in "Forum."

Definitions

Let's start by defining a few basic terms. It has been said that to understand all the words means to understand the subject.

CRT (Cathode Ray Tube)

This is the central element in all but the most recent video display systems (i.e. pocket TV's based on LCD's or "Liquid Crystal Displays."). The term "CRT" derives from the very infancy of electronics, when it was discovered that certain rays were emitted by the cathode (negative terminal) in a tube from which most of the air was evacuated (i.e., a "vacuum tube.") These particles were later named "electrons," and are in fact what electronics is all about. Still, the original terminology has stuck to this day.

A monochrome CRT has at its heart an "electron gun," which is the heated cathode from which electrons are emitted. These are formed and focused into a narrow beam by various electrodes known as "grids" (a throwback to the nomenclature of vacuum tube technology). One of the grids is also used to vary the intensity of the beam. (This is important both for setting the average brightness with the Brightness control, and also the instantaneous brightness changes which make up the final picture.) The beam impinges on the front of the CRT (the part you look at), which is coated with a "phosphor" which emits light when struck by electrons.

A color CRT is similar, except that it has three separate electron guns; one for red, one for blue, and one for green. (Using light sources in these three primary colors allows us to make any other color by appropriately blending them.) The face of the CRT is covered with dots of phosphor instead of a continuous layer. Right behind the face is a "shadow mask," or plate with 100's of thousands of tiny holes. It is carefully positioned such that the electron beams from the three different guns strike the appropriate color of phosphor dot. The key to operation is parallax; through any given hole, a red, green, or blue dot can be lit by "firing" the appropriate electron gun.

Around the neck of the CRT are a set of "deflection coils," which are electro-magnets that control the position of the electron beam(s). These coils are connected to the outputs of the vertical and horizontal "oscillators," which cause the electron beam to sweep back and forth, and up and down.

Video

The term "video" is the first-person singular of the Latin verb videre, "to see." (The title of the Vol 1 article thus translates as, "I came, I saw video, I conquered!") In electronics, "video" means "relating to the electronic production of visual images." More specifically, it is the name given to the signal causing the electron beam(s) in a CRT to vary in intensity, making the light and dark (or colored) areas in the final picture. Note that for monochrome, only one video signal is required, whereas for the simplest color systems, THREE separate signals are needed: one for each primary color.

Synchronization

Most of us know what it means to synchronize something. The word comes from Greek roots meaning "same time." As relates to video systems, this refers to signals that mark exactly when each frame (picture) starts, and when each line within the frame begins. These "sync" (short for "synchronization") pulses cause the vertical and horizontal oscillators in the destination device (your TV or monitor) to "lock step" with the corresponding signals in the source device (video camera, VCR, or computer).

Composite Video

Let's say, in a monochrome video system, that we wish to run only one wire (or use only one band of radio frequencies) to convey all the video information. However, we have three discrete signals to deal with; vertical sync, horizontal sync, and video. These can be combined into a single signal, appropriately called a "composite" video signal. See the SWN Vol. 1 article for a more detailed description of how this is done.

In the case of color, the problem is even hairier. We have not three, but FIVE different "primal" signals to deal with; the two sync signals, red, green, and blue. I used the term "primal" in the last sentence, because there are several more intermediate signals generated in the mix-down (and later, in the decoding) processes. I'll state without proof that a composite color video signal is very complex. I'll also make the somewhat philosophical statement, that it's amazing that composite color works at all, truly miraculous that it works as well as it does.

Luminance & Chrominance

The video information in a composite color system is broken down into two basic elements, "luminance" (Latin, "lumen" = "light") and "chrominance" (Latin, "chroma" = "color"). Luminance simply relates to "how much light," whereas Chrominance means "how much color." Luminance therefore describes the monochrome signal, whereas chrominance additionally specifies the colors of that signal.

Why bother with luminance at all? After all, given the color information, we should be able to control intensity as well. The answer is akin to the reason that color photographs contain silver (black) in addition to the colored dyes. It results in better control of the finished picture. Your "contrast" control varies luminance, whereas "color" (chroma) varies the chrominance.

How do we get three colors from one chrominance signal? The full answer is far beyond the scope of this article. Suffice it to say that this is done by varying the timing of the chroma signal, relative to a signal called the "color burst" or "chroma burst" sent by the source, and duplicated at the destination by an accurate crystal. The relative phase (timing) of the chroma signal determines the color. You can adjust the average (or quiescent) phase by adjusting your "tint" control.

Monitor

We all know what a TV is, so I won't waste space on it here. But how does a "monitor" differ from a TV? The answer is similar to the difference between a radio and an audio amplifier/speaker. As I've said before, a TV is simply "a monitor with radio." A television set contains a tuner (to receive radio signals encoded with video information), plus "intermediate frequency" amplifiers and a "detector" to convert the radio signal into a composite video signal. (We'll ignore the audio portion here.)

Many TV's are being sold these days with a "monitor input." The manufacturers usually allude that this feature is what justifies a considerably larger price tag. In reality, they've had to do very little more than install a jack and a switch to a standard television. If you are looking for a monitor which doubles as a TV set, you might be better off buying a real composite monitor and a (perhaps used) VCR; to watch TV, simply use the tuner in the VCR. NOTE: most home VCR's will NOT work with RGB monitors.

RGB

As you probably know, "RGB" simply means "Red, Blue, Green." What you probably don't know, is that an RGB monitor is the SIMPLEST color video system you can get. Three separate lines (R, G and B) are used to directly control the three electron guns, and an additional line or lines supply the sync signals. Again, manufacturers would like you to believe that there is something very special about RGB, justifying a larger price tag. Not so. Part of the explanation of higher cost has to do with production volumes, the other part is that RGB looks great, and therefore nobody complains. So why is RGB the simplest, and why does it look so good? There's a

lot less electronics that the signal has to go through before reaching the CRT, since the monitor doesn't have to decode the composite video signal into sync, luminance, and chrominance elements. The computer doesn't have to laboriously assemble the R, G and B signals into composite, only so that the composite monitor can take it all apart again before sending it to the electron guns.

Resolution

Resolution is a measure of how many discrete dots or lines can be displayed before they merge together. In photography, the term is very precise, and is usually expressed in "lines per millimeter." In video, it's usually much more qualitative. Monitors generally break down into "low," "medium," and "high" resolution, with corresponding quantum jumps in price.

Horizontal

Resolution also has different meanings when applied to monochrome or color monitors. In a monochrome unit, resolution depends almost completely on only two factors; dot size (diameter of the beam from the electron gun), and video bandwidth. The effect of dot size on resolution should be quite obvious; the smaller the dot, the higher the resolution. Bandwidth describes the maximum frequency (and therefore, discrete dots per line) that the video amplifier is capable of. Low resolution is anything up to around 4 mHz. (mega Hertz, formerly megacycles per second). Medium resolution is around 4-7 mHz., and high resolution is anything above that. Standard NTSC (American) television has a bandwidth just under 4 mHz., a limitation due to the restricted space allowed in our television channels. (This is one reason why a TV with "monitor" option is usually not as good as even an inexpensive composite "true" monitor; they are designed as TV's first, monitors second.)

In color monitors, a third factor affects resolution. This is that the phosphor is made of discrete dots corresponding to the shadow mask, instead of a continuous sheet. On the low-res CRT's used on televisions, the number of dots (dot resolution) was chosen to be just within the bandwidth limitation. By minimizing the number of holes in the shadow mask, costs are kept down. So, even if "TV with monitor option" sets have video amps with a larger bandwidth, the CRT will itself impose a resolution restriction.

The definition of high, medium, and low dot resolution depends on who you talk to. Perhaps the best definition is in the QL manual, and I quote:

"The horizontal resolution of the QL is 512 pixels across the screen. This means that a monitor used with the QL should have a shadow mask resolution adequate to cope with this. For a 14-inch monitor, this resolution should be at least one dot every 0.43 mm. (known as "medium resolution" in England, and "high resolution" in the U.S.); one dot every 0.31 mm. ("high resolution" in England) is better still. Monitors with standard TV tubes (one dot every 0.85 mm.) are usable, but can suffer from poor distinction of vertical lines."

The above implies, that as regards our TS1000 and TS2068 machines with 256 horizontal dots, the low-

res TV-type monitors are entirely adequate. And indeed they are, for the most part. When dealing with 64-column screens, however, they are woefully inadequate. For such text applications, you're MUCH better off with even a cheap monochrome monitor.

Vertical

Vertically, resolution is less of a problem. This is because our machines (TS1000 family, and Spectrum/TS2068) use "non-interlaced scanning." This requires an explanation of "interlacing." In video as used by television, the horizontal scan rate (and therefore sync signal) is a non-integral multiple of the vertical rate. More precisely, it's an integral multiple of HALF the frame rate, or 30 Hz. The result is that the two sets of lines from subsequent 1/60th second frames, will "interlace," i.e. the second set of lines falls between the first set.

This brings up a point of semantics. We computerists use the term "frame" to specify the 1/60 sec. interval during which an entire screen is displayed. This is more precisely termed a "field." Two fields (1/30 sec.) make up one "frame," or complete TV picture. (Comes from a "frame" of a motion picture film, but the analogy is not exact since motion pictures usually run at 24 frames per second).

Some computers DO use interlaced scanning, but the TS family does not. You might think of this as using only the odd-numbered lines available on the TV or monitor. This is why, on monochrome monitors, horizontal lines appear as lines, whereas vertical lines appear as row of dots. In still other words, our "frame" (complete picture) is the same interval as our "field" (vertical scan interval), so it is correct to refer to the 1/60 second vertical interval as a "frame."

R.G. Blues

The RGB system is not standardized from one manufacturer to another. Some RGB monitors have a single sync line carrying both horizontal and vertical pulses (composite sync), whereas others have separate sync lines. The QL gets around this rather cutely by supplying a composite sync and a vertical sync; monitors that require separate sync can usually live with the composite sync instead of horizontal.

There isn't perfect agreement about the phase of the sync signals. This is especially a problem in European monitors (and computers, as the QL or the Amstrads) vs. American ones. Frequently, one or the other or both sync signals will need to be inverted with a gate such as a 74LS04. If your RGB monitor simply won't "lock in" your picture (rolls or horizontal tearing that can't be stably adjusted with the "hold" controls), you will probably only have to invert the offending line to cure the problem.

Another thing we have to cope with is "analog" vs. "TTL." "Analog" means that the signals can be varied in intensity, just as in standard video. "TTL" means that any given dot is either on, or off; nothing in-between. For 16-color computers as the Sinclair family, the TTL type is superior (if your RGB interface supports it). However, other machines (as Amiga and ST) require the analog type, so that they can

reproduce the nuances of 256 (or more) colors by varying the intensity of the dots.

A feature on some TTL monitors is an "intensity" input. This basically allows the monitor to be used in an analog mode. Think of it as being a luminance input, separate from the "discrete chrominance" of the R, G, and B inputs. If your present computer doesn't support this input, it generally does no harm to simply leave it open.

More Info

I'll close with suggestions for buying the right monitor for the job, and a few other side notes.

If you're planning on staying with the TS machines, a "standard" (i.e. low-res) monitor is entirely adequate. However, if you're later planning on getting a more advanced machine, such as QL, Atari ST, or Amiga, a higher resolution monitor bought now will save you money later when you really need it.

If you're mainly interested in text applications, by all means get a monochrome monitor. Not only is it a lot cheaper, it is also a lot cleaner.

Some users may want a monochrome monitor for their day-to-day serious applications, but would like to play the occasional game or see graphics programs in color. You don't HAVE to buy a color monitor! If your computer has a TV output (as 2068), use that for those "rare" occasions. If not (as Oliger TS1000, some Atari ST's, probably others), get a modulator from Radio Shack. Alternately, run your composite video into the "Video IN" jack on the VCR.

If you want the easiest-to-use color monitor, get one with composite video. It'll plug right in, on the 2068 or "Oliger 1000." You don't have to worry about the interfacing hassles that can plague RGB systems.

If you want the best possible picture, the ONLY way to go is with RGB. Though it may rub you the wrong way to pay more for less (it does me,) "facts is facts" and nobody guaranteed life to be fair.

Beware of those "\$19 monitor bargains." Though they may be excellent monitors, these are usually designed for some specific application (such as surveillance). These are neither composite nor RGB, and won't work with either type of system. Many such monochrome monitors have separate sync and video inputs. You might get these to work with a circuit as described in the first "Primer." There are also color monitors with separate luminance and chrominance inputs ("separated video"); we haven't found a simple way to deal with these, so they are best avoided.

Sound is a nice feature on a monitor, even if your present system doesn't support it. Who knows what you might buy later on? Note that the sound input on most monitors requires a line-level (about 2 volts) sound input. The 2068's MIC output, of course, is much lower; around .0001 volts. The level at the edge connector 31A is not much higher. So to use

this with your monitor, try tapping it into the output of the sound op amp output (actually part of the switching regulator), at pin 4 of U1, or at the speaker (junction of CR24 and R19).

How big should your monitor be? Though it's largely a matter of taste (and eyesight), a few subjective hints may help. My favourite monitor for word-processing is a 9" green screen, positioned about two feet from my eyes. Provided that you don't sit too far away, a smaller monitor actually gives a sharper look. I've done a lot of programming on my "compact 1500 system," which has a 3" screen. When working on this system, my eyes are about a foot from the screen; under such conditions, that tiny display is entirely adequate (after all, it's about the same size as a 2040 listing). For playing games and goofing with graphics, I prefer my 13" composite color set. Our 21" TV is ok for games only if I'm about 10 feet away from the screen (and even then it's dicey).

WINKJET1 Review

Robert D. Hartung

If you saw the reviews of the PR2300 when it was introduced several years ago, you know that the original list price was about four times more than it was finally sold for by various discount houses. The reviews also rated some of its features as being well worth the price, such as outstanding graphic capabilities, fast and quiet operation, good quality output, and rugged construction. A unique feature is that instead of print-wires being driven against a ribbon as in other dot-matrix printers, an electrical charge fires dots at the paper from economical ink ampules. When an ampule is replaced it is the same as having a new printing head.

After discontinuing production, Olivetti has been less than helpful with documentation. Since neither the T/S computer nor the PR2300 closely follow the nominal standards more or less accepted by a major part of the industry, many have experienced considerable frustration in trying to realize the potential of mating the two, particularly in the plotting of graphics. I do not personally own a PR2300 but have spent some time using one. After an entire afternoon of trying to adapt the example listing shown in the manual so it would plot two tiny triangles, I finally discovered there was an error in their listing. Fortunately, there is now help on the horizon in the person of William J. Pedersen of the WIDJUP Co. A retired aerospace engineer, he has unraveled the secrets of the PR2300 (and also has developed some unique bank-switching techniques for the TS2068). He has a large catalog of other utility programs for TS1000/2068s.

Pedersen's WINKJET 1 is designed specifically for the PR2300/TS2068 combination. A relocatable 157-byte printer module is included, with pre-defined options for the TASMAN A and B and AERCO interfaces. Provisions for customizing it to any other I.F. as well as output to the TS2040 or Alphacom printers are also included. After loading the program, prompts guide you step by step through the initiation of the module for your interface. This data is saved and may then be merged and used with other programs such as those needed to produce high-resolution drawings and graphics or text on the screen to be copied.

Other features to look for:

- 1: Easy adjustment of horizontal width (almost crucial for TI video and QL).
- 2: DC Restoration switch (composite & Mono)
- 3: 75 ohm / Hi-Z switch (composite)
- 4: Desired monochrome color (amber, green or white)
- 5: Reversing switch (monochrome)
- 6: Front-panel controls (bright, contrast, v/h hold, color & tint if applicable).

Hope this helps. If you just jumped out of your chair yelling "Nunc video!" (Now I see!), then my purpose is complete. If not, let us know. We also welcome reader suggestions on getting around RBG-related problems, adapting "bargain" monitors with unusual video/sync requirements, and anything else you may have discovered.

WINKJET supports LPRINT to drive your printer directly, using the resident font, and makes it easy to access all PR2300 commands, as any printer driver should. Its greatest strength, however, lies in the ability not only to copy exactly whatever is on the screen but to selectively scan the DFILE1 (and DFILE2 if it is in use) by means of user-defined "windows." It will then send data in proper order to the printer for it to graphically plot to paper whatever it finds there, whether it be normal T/S characters, user-defined fonts, UDGs, graphic plots, or any combination of these.

These windows may be used to correctly position up to the maximum of 110 characters per line, or to scan a single row of pixels at a time as set by your PLOT, DRAW, or UDG routines for the highest resolution obtainable from the PR2300, in normal or inverse image. This all may be done in exact copy size, double width or double height text or both, or ZOOM plotting mode. Resolutions in a range from 1 to 9/216 inch are provided by the windows. The option of double scanning produces a dot density that is almost a solid black in plotting a display area in which all the pixels are set.

WINKJET 1's greatest usefulness is as a tool to communicate to your PR2300 what you want done with what you have put into the display file. For this reason, it requires and is easily merged with a routine of your own that puts printed characters and/or plotted graphics to the screen. To use it, GO TO or GO SUB is then made from main program to the pre-defined window routines or to any you customize (in Basic) with your own definitions. This may seem to be a lot of footwork, but if you have ever tried to do graphics another way you know good things don't come easily.

WINKJET 1 for the TS2068 is available for \$14.95 + \$1.50 S & H from WIDJUP Co., 1120 Merrifield S.E., Grand Rapids, MI 49507. Versions are also available for use with Star and Epson type printers as well as the Seikosha 100, Panasonic JR-PO2U, and the Gorilla Banana. (This provides an easy fix for those short-descender characters IF you can live with the fact that plotting mode MUST be used, which will

make an exact or enlarged screen-dump copy of whatever you put in the display). Included is a program called TypaLot which is designed as a simple, no-frill, 64-column word-processor to either LPRINT directly or put text on-screen when merged with WINKJET 1. It supplies two additional type fonts (script and enhanced T/S) or you may define your own if you want to. It also prints sideways on Alphacom/TS2040 printers.

PUT MSCRIPT ON EPROM

by David W. Lockin

MSCRIPT is a very useful program. This article is for improving MSCRIPT, although it does not change any of the coding. What we are going to do is make it load faster by transferring it to an EPROM.

The following for putting MSCRIPT on an EPROM is only for the original tape version. It will not work on Jack Dohany's improved and customized versions.

This article is for the hardened 2068 user. It requires that you have, or can lay your hands on, the following: one unused 8K X 8 EPROM (2764), one cartridge board (John Oliger or equivalent), one 2068 compatible EPROM programmer (John Oliger or equivalent). One more item that is helpful, but not necessary, is a copy of HOT-Z 2068.

You will be entering data from tables 1,2, and/or 3. For the normal version of MSCRIPT, enter data from only tables 1 and 2. If you want fat characters, as in SyncWare News 2:4, then enter the data in tables 1 and 3 only. Also, the data at address 36628 (8F14h) must be changed from 34d to 61d for the fat characters. This point has been marked with an "*" in TABLE 1.

Data is entered from the tables left to right and top to bottom. If you will be using the Tasman version of Mscript, there are special instructions for the TASMAN INTERFACE VERSION at the end of this article.

If you have HOT-Z (especially HOT-Z AROS), I recommend using it at this point. Load HOT-Z. Load the machine code portion of MSCRIPT into HOT-Z. Then enter either assembly listings 1&2 for normal characters or assembly listings 1&3 for the fat character version. Remember to include the modifications for the TASMAN Interface if you are using it. Jump on ahead to the BURNING YOUR EPROM section.

If HOT-Z is unavailable to you, the first thing to do is to load your MSCRIPT program. Load it using the following comand:

```
LOAD "CODE
```

Load only the machine code portion. Do Not load the basic loader or the screen logo.

```
"G127;32;22;20Z";
```



WINKJET SAMPLE PRINTOUT

Next, enter the following basic poker program. This program was borrowed from the Tom Woods' BREAKTHROUGH newsletter.

```
10 FOR X=36608 TO 36668 (36694 FOR FAT CHARACTERS)
20 INPUT (X;"=");Y$
30 IF Y$="" THEN GOTO 60
40 IF LEN Y$=5 THEN LET X=VAL Y$: PRINT X;"=";PEEK X
GOTO 20
50 POKE X,VAL Y$
60 PRINT X;"=";PEEK X
70 NEXT X: STOP
```

Save this program. If anything goes wrong you can use it to re-check your code.

To run this program use GOTO 10. You will see a number at the bottom of the screen. This number is the address of the data that you will enter into the program. Hit enter after you type in each data value and it will move to the top of the screen. The next address will appear at the bottom of the screen.

Remember, for those of you who are going to use the fat characters, the following data has to be changed: the data at addr 36628 has to be changed to 61d. Again, this point has been marked with an "*" in TABLE 1.

Burn Your EPROM

I suggest that you go back and check the code you just entered. This is easily done by flipping the pages in HOT-Z. It's just as easily done with the basic poker program too. Enter GOTO 10 again. Now instead of entering data, you just hit enter. This will show the address and its data. Check them against the tables.

The code you just entered is the AROS I.D. code and the program transfer code. The first 8 bytes are the AROS I.D. code. The rest is the transfer code (and fat character code if included).

If you were using the basic poker, save your new code and MSCRIPT as one block of code. Do this by using one of the following two commands:

```
SAVE "ESCRIP"CODE 36608,8192
SAVE "FATESCRIP"CODE 36608,8192
```

To save from HOT-Z, set the cursor to 8F00 and END to AEFF.

Now that you have protected yourself from an unforeseen disaster, we will get on to the actual programming of the EPROM.

Before we start programming the EPROM, install any other modifications to MSCRIPT you desire. Once the code is in EPROM, the only way you can change it is by burning a new EPROM. If you make any other changes to MSCRIPT, save the entire program to tape as already described when finished.

The following is based on using the Oliger EPROM programmer. If you have a different programmer, its program may be different, but everything else will be the same. If you have the most recent version of HOT-Z AROS, then you have the capability to burn your EPROM from HOT-Z. Just follow the instructions in the documentation. Enter the following basic program unless you're using the above mentioned version of HOT-Z:

```
10 LET X=8192
20 FOR N=36608 TO 44800
30 POKE X, PEEK N:PRINT AT 10,11,N
40 LET X=X+1:PAUSE 3:NEXT N
50 BEEP 1,1
```

ADDR	DATA	ADDR	DATA	ADDR	DATA	ADDR	DATA	ADDR	DATA
36608	2	36609	2	36610	8	36611	128	36612	15
36613	1	36614	0	36615	0	36616	243	36617	62
36618	240	36619	211	36620	244	36621	33	36622	27
36623	128	36624	17	36625	0	36626	112	36627	1
36628*	34	26629	0	36630	237	36631	176	36632	195
36633	0	36634	112	36635	1	36636#	0	36637	32
36638	17	36639	0	36640#	144	36641	33	36642	0
36643	129	36644	62	36645	48	36646	211	36647	244
36648	126	36649	245	36650	175	36651	211	36652	244
36653	241	36654	18	36655	35	36656	19	36657	11
36658	175	36659	184	36660	32	36661	238	36662	185
36663	32	36664	235						

TABLE 1

ADDR	DATA	ADDR	DATA	ADDR	DATA	ADDR	DATA
36665	251	36666	195	36667	0	36668	144

TABLE 2

ADDR	DATA	ADDR	DATA	ADDR	DATA	ADDR	DATA	ADDR	DATA
36665	17	36666	0	36667	88	36668	213	36669	1
36670	3	36671	42	36672	54	36673	92	36674	36
36675	126	36676	167	36677	31	36678	182	36679	18
36680	35	36681	19	36682	13	36683	32	36684	246
36685	16	26686	244	36687	225	36688	37	36689	34
36690	54	36691	92	36692	195	36693	0	36694	144

TABLE 3

Zap It

Now save this program. Once it is saved, turn off the computer. Connect the programmer board with EPROM installed to the computer. Connect the programmer power supply and set it to 4.5V. Then turn the computer back on and reload your new MSCRIPT program and the programmer program. Enter the following command: GOTO 10. Switch the programmer voltage to 21V, then hit ENTER. Now sit back and relax for about 12 minutes. The program will beep when it is done. Turn the programmer supply to 4.5V and turn off the computer. Turn off the programmer supply and remove the programmer and the EPROM. Install the EPROM in a properly prepared cartridge board. The EPROM has been programmed to run from 8000H to 9FFFH.

Now comes the big test. Plug your cartridge board into the computer and turn it on. MSCRIPT should now be up and running. If it is not, you did something wrong and you will have to go back and recheck your work.

You are now finished. I hope you have a working copy of MSCRIPT on an EPROM. I realize that this may seem like a lot of work, but I feel anything that can make using a program easier is worth while. I believe that MSCRIPT is worth the trouble. If you do

```
8F00 02 DATA
8F01 02 DATA
8F02 08 DATA
8F03 80 DATA
8F04 0F DATA
8F05 01 DATA
8F06 00 DATA
8F07 00 DATA
8F08 F3 DI
8F09 3EF0 LD A,F0
8F0B D3F4 OUT (F4),A
8F0D 211B80 LD HL,801B
8F10 110070 LD DE,7000
8F13 012200 LD BC,0022(003D)
8F16 EDB0 LDIR
8F18 C30070 JP 7000
8F1B 010020 LD BC,2000[2072]
8F1E 110090 LD DE,9000[8F00]
8F21 210081 LD HL,8100
8F24 3E30 LD A,30
8F26 D3F4 OUT (F4),A
8F28 7E LD A,(HL)
8F29 F5 PUSH AF
8F2A AF XOR A
8F2B D3F4 OUT (F4),A
8F2D F1 POP AF
8F2E 12 LD (DE),A
8F2F 23 INC HL
8F30 13 INC DE
8F31 0B DEC BC
8F32 AF XOR A
8F33 B8 CP B
8F34 20EE JR NZ,8F24
8F36 B9 CP C
8F37 20EB JR NZ,8F24
```

LISTING 1

```
8F39 FB EI
8F3A C30090 JP 9000
```

LISTING 2

```
8F39 110058 LD DE,5800
8F3C D5 PUSH DE
8F3D 010003 LD BC,0300
8F40 2A365C LD HL,(chas)
8F43 24 INC H
8F44 7E LD A,(HL)
8F45 A7 AND A
8F46 1F RRA
8F47 B6 OR (HL)
8F48 12 LD (DE),A
8F49 23 INC HL
8F4A 13 INC DE
8F4B 0D DEC C
8F4C 20F6 JR NZ,8F44
8F4E 10F4 DJNZ,8F44
8F50 E1 POP HL
8F51 25 DEC H
8F52 22365C LD (chas),HL
8F55 C30090 JP 9000
```

LISTING 3

not want to invest in a disk system, then putting your most used programs on a cartridge is worth the bother. If this helps in making your computer a little less of a hassle to use, you may use it more. In using it more, you may find that it wasn't such a bad investment after all.

Tasman I/F Notes

For those of you with the Tasman interface, the major difference is that it will not fit on one 2764 (8K X 8) EPROM. You will need two 2764's or one 27128. The following instructions will be based on programming a 27128 EPROM. Enter the previously described basic poker program but change line ten to read:

```
10 FOR X=36352 TO 36412 (36438 FOR FAT CHARACTERS)
```

Enter the data as shown in tables 1&2 for normal characters. If you want fat characters, enter tables 1&3. The data in TABLE 1 must be changed at the following addresses: 36636=114 (8F1Ch=72h) and 36640=143 (8F20h=8Fh). These locations are marked with a "#". For fat characters, also change the value at 36628 (8F14h) to 61 (3Dh). The starting address for the TASMAN listing is 8E00. This is to allow for the interface code that has been added at 8F00. Once the data has been entered and checked, save your new code and MSCRIPT as one block of code. Do this by using one of the following two commands:

```
SAVE "ESCRIP"CODE 36352,8572
SAVE "FATESCRIP"CODE 36352,8572
```

From HOT-Z, set the CURSER at 8E00 and END to AF72.

Before you start programming the EPROM, install any changes or modifications you wish. Resave if you make any modifications.

Enter the following program:

```
10 RESTORE data: READ src: READ end: READ dest:
   READ one: READ three: READ loop: READ line:
   READ col
20 POKE dest,PEEK src
30 PAUSE three
40 LET src=src+one
50 LET dest=dest+one
60 PRINT AT line,col;dest
70 IF src<=end THEN GOTO loop
80 DATA 36352,44924,0,1,3,20,10,13
```

Save this program. Once saved, turn off the computer and connect the programmer board with the EPROM installed to the computer. Connect the programmer power and set it to 4.5V. Turn the computer back on and reload your new MSCRIPT program and the programmer program. Enter the following statement:

```
LET p=10: LET data=80 Press enter.
```

Make sure the 64/128 switch is in the 128 position. Set the programmer supply to 21V and key in GOTO p and press enter. Now sit back and relax for about 12 minutes. When the programming is done, turn off the programmer supply and turn off the computer. Remove the programmer and the EPROM and install the EPROM on a properly prepared cartridge board. The EPROM has been programmed to run from 8000H to A200H.

Test the EPROM assembly as directed above.

MINI BOOK REVIEW

Best of PCW Assembler Routines

Edited by David Barrow

Reviewed by Fred Nachbaur

Correspondent Larry Chavarie sent me a copy of this book for review. He had sent his order to Personal Computer World, a prominent British computing magazine, who then forwarded it directly to the author/editor to be filled. It didn't take me long to realize that this book is a real gem. The back cover panel summarizes it best:

"Every serious assembler programmer needs well-documented sub-routines. This is a collection of the best routines for the Z80 compiled from the famous SUBSET series that has been running in "Personal Computer World" for several years....

"This book presents the distilled experience of the best machine-code programmers, and will be an essential part of any serious assembler programmer's library."

The routines are organized into chapters, starting with the simple (delays) and progressing to the highly sophisticated (extracting roots and 32-bit arithmetic). Along the way, topics covered are comparisons, searches, sorting, data compression, graphics, conversions, random numbers, and matrix transposition; to name but a few. All routines were written to strict standards, set out in a series of rules published at the onset of the series. Included in these standards is the rule that no registers be corrupted (except those used to pass parameters). Each routine includes a "datasheet," which is a 14-point summary of the routine's characteristics. Every line is annotated. Additional descriptive text makes the documentation as complete as anyone could possibly want.

I agree with the writer of the back panel. This is a "must have" for serious programmers on any Z80-based machine. Cost is 7.95 (pounds sterling); be sure to include enough for return shipping. Order directly from:

David Barrow
9 Dorset Close
Pontefract
West Yorkshire
WF9 4HU
United Kingdom

BASIL'S COMPENDIUM

Basil Wentworth

This chapter will tell you about conditional instructions, and will discuss the behavior of various flags.

IF... THEN

If I was asked to name the single most powerful attribute of a computer, I think that I would say it's the ability to make decisions--to vary performance with varying conditions.

Conditional instructions in machine code can take the following forms, with the analogies of BASIC given in parentheses:

1. IF...THEN JUMP; or IF...THEN JUMP RELATIVE (GOTO)
2. IF...THEN CALL (GOSUB)
3. IF...THEN RETURN (RETURN)

The respective instructions for these operations are listed in Table 12-1. The "ifness" of the situation is measured by the conditions of various flags. You will remember that the flags reflect the results of the most recent operations of the program.

A sample situation calling for a conditional JUMP would be as follows: what is the largest number that is a power of two, but is not greater than 1000? A BASIC solution to this problem is shown in Figure 12-1. If you can't visualize how this program works, then put it into the computer and RUN it. (Adjust the line numbers so it won't

```
1010 LET N=2
1020 LET N=2*N
1030 IF 2*N>=1000 THEN GOTO 1050
1040 GOTO 1020
1050 PRINT N
```

Figure 12-1. The solution in BASIC

```
16514 33 5 LD HL,2
16515 2 Gr 2
16516 0
16517 17 )
16518 232 CONT
16519 3 Gr 7
16520 68 ? LD B,H
16521 77 ? LD C,L
16522 41 D ADD HL,HL
16523 237 GOSUB SBC HL,DE
16524 82 ?
16525 208 SQR RET NC
16526 25 ; ADD HL,DE
16527 24 / JR,-9 **
16528 247 RUN
```

**NOTE that JP 16520 could have been used instead of JR,-9

Figure 12-2. The machine code approach

```
1010 LET N=2
1020 LET A=1
1030 LET N=N+2
1040 LET A=A+1
1050 IF A=3 THEN GOTO 1070
1060 GOTO 1030
1070 PRINT N
```

Figure 12-3. 2*3 in BASIC

```
LD A,3
LD H,1
*ADD BC,2
INC H
CP H
JR NZ,...or JP NZ...
RET
```

*There is no instruction for ADD BC,2. Try INC BC two times.

Figure 12-4. The mnemonics alone

interfere with any program material you already have there, and be sure to delete those lines when you've finished, to avoid possible trouble later on.)

Figure 12-2 shows how to do the same thing in machine code. Notice that the RETURN (201) is not needed, PROVIDED:

1. that some other form of RETURN instruction, such as RET NC, is included, AND
2. that the conditions for RETURN will indeed be met during the running of the program.

WATCH OUT there! Both conditions must be met. It's oh so easy to go to great pains to meet condition 1, and then forget to make sure that condition 2 is met.

It may seem a little bizarre to subtract DE from HL and then add it right back again, but that's what line 1030 of the BASIC program did in Figure 12-1. However, you can avoid this extra step by using the instruction COMPARE (mnemonic: CP).

Compare

CP makes the comparison without actually changing the contents of the register being compared. This is a very useful instruction, and it can save you a lot of grief in some situations, but it is subject to two limitations:

1. CP will compare a specified quantity with the contents of Register A only; and
2. therefore, you can compare only a single-byte quantity (0-255) in a single step.

```
16514 62 Y LD A,3
16515 3 Gr 7
16516 38 A LD H,1
16517 1 Gr 1
16518 1 Gr 1 LD BC,2
16519 2 Gr 2
16520 0 space
16521 3 Gr 7 INC BC
16522 3 Gr 7 INC BC
16523 36 8 INC H
16524 188 Inv W CP H
16525 32 4 JP NZ,-6
16526 250 IF
16527 201 TAN RET
```

Figure 12-5. The machine code for Fig. 12-4

Of course, you can compare a two-register variable in two successive steps, if you're willing to give it a little thought. See if you can figure out how to do it after you read the following few paragraphs. If you're still stumped, the ANSWER section at the end of this chapter will help you.

To watch CP at work, let's set up a very simple multiplication problem. Again in BASIC, use the program shown in Figure 12-3. Do you see how this program multiplies 2*3? (Again doing it the hard way, but never mind that for a moment.) Now, how would you do the same thing in machine code?

Figure 12-4 gives a possible set of mnemonics for the solution. Try working out the code for yourself, and compare your solution with that shown in Figure 12-5. Your solution may be somewhat different. There's room for individual style in machine code, just as there is in BASIC. You may even hit on an entirely different approach. If it works, it's good. If it is shorter than mine, or runs faster, or is easier to understand, then it's more elegant.

Always try for those three objectives: economy in use of memory, speed of operation, and clarity. The first two usually go hand-in-hand. Usually, but not always. If the three objectives conflict with one another, you'll have to make a judgement as to which is the most important in the application you have in mind.

Flags

Figure 12-6 gives a little program that will help you to find out what conditions SET and RESET the various flags (i.e., make them equal to 1 or 0, with 1 meaning SET). For the moment, you'll be concerned with three flags: S (=1 if the last operation resulted in a positive number, =0 if the result was negative); Z (=1 if that result was zero, otherwise =0), and C (=1 if the operation resulted in a carry).

The S, Z, C information is carried in the F register, in the following manner:

S: in the 128 byte
Z: in the 64 byte
C: in the 2 byte

The program itself lies in lines 1-95. The rest of the lines can be omitted, if you use the EL MINIMO

Condition:	ALL	Z	NZ	C	NC	P	M
JP,NN*	195	202	194	218	210	242	250
JP(HL)	233						
JR,DIS	24	40	32	56	48		
CALL,NN*	205	204	196	220	212	244	252
RET	201	200	192	216	208	240	248

*NOTE: NN refers to a two-byte piece of data, LSB first

Table 12-1. The conditional instructions

system or any other means of writing your machine code into line 1. Lines 510-530, by the way, serve the purpose of the FLASH instruction that is found on some other computers. It's a spectacular way of attracting attention.

The five bytes already shown in line 1 (represented by PRINT, AT, Gr T, space, and TAN) stand for

```
PUSH AF (245)
POP BC (193)
LD B, 0 (6,0)
RET (201)
```

These five bytes make USR 16514 calculate the value held in Register F. The spaces in front of this sequence are nulls, to be filled in with whatever operations you want to test. Zeroes are used for these nulls so that they won't interfere with the operation of the program if one or more of them is inadvertently left in.

Answer Section

If you weren't able to work out the answer to the problem posed above (use CP to find whether X is greater than 1000), try this:

1. Let's assume that X is stored in register BC, so that $X = C + 256 * B$.
2. Is B greater than 3? If so, then X is greater than 1023.
3. Is B less than 3? If so, then X is less than 768.
4. If B=3, then is C greater than 232? If so, then X is greater than 1000.

That's all for now. There will be more.

```

1 REM Y COPY W PRINT AT " T
AN
5 FAST
10 REM DISPLAY FLAGS
15 PRINT
20 LET N=USR 16514
30 LET S=INT (N/128)
40 IF NOT S THEN PRINT "- "
50 IF S THEN PRINT "+ "
60 LET N=N-128*S
70 LET Z=INT (N/64)
80 IF Z THEN PRINT "Z "
90 IF 2*INT (N/2)<>N THEN PRINT "C"
T "C"
95 GOTO 9000
100 REM INPUT CODE
105 FAST
110 CLS
120 PRINT "BE SURE THERE ARE ENOUGH NULLS FOR YOUR MACHINE CODE"
130 PRINT
140 LET A=16514
150 PRINT A;" "
160 IF PEEK A=118 THEN GOTO 500
170 INPUT B
175 IF B>255 THEN GOTO 9000
180 POKE A,B
190 PRINT PEEK A
200 LET A=A+1
210 GOTO 150
500 SLOW
510 PRINT AT 10,10;"NOT ENOUGH NULLS"
520 PRINT AT 10,10;"NOT ENOUGH NULLS"
530 GOTO 510
8999 STOP
9000 REM 0000000000000000
9020 PRINT
9030 LET F=16514
9040 PRINT F;" ";PEEK F
9050 LET F=F+1
9060 IF PEEK F=118 THEN STOP
9070 GOTO 9040

```

FIGURE 12-6. THE FLAGS

CLASSIFIED

Do you have something to sell?
Let everyone know! Just \$2.75 a
line will put it here. 32 column
(screen) format, include spaces
in your text, and send CK or MO
with your listing.

Send to SWN Classified, 602 S.
Mill St., Louisville, OH 44641

Do color art AND scaled tech
drawings with 2068 and ACCUDRAW!
Distance and angle readouts, two
display files, two undos, four
screen dumps, rubberbanding, 2-
speed cursor, fill, mirror, &
more. Use desk mount stick with
thumb button or contact mouse.
Best with Aerco CIF. \$19.95 (+
1.05 CA tax)(includes S&H) M.O.,
check, or SASE to GLENN Technics
Box 2760, Anaheim, CA 92084

TX1000 mounted in keyboard. Forth
eprom, UHF 31, load/save monitor,
BB modem/64K ram, 2050 modem, 2040
printer, cassette recorder, TS 16K
ram, Profile, Chess, Hot Z, Pilot,
Toolkit, Z80 manuals, SYNC mags.
\$400. for all. (303) 972-8726 Doug

Now, we know that you like contests because our
first annual affair was a smashing success. So why is
it that nobody's entering our second annual contest?

Actually, that's not quite true, at press time, we
have 3 entries. That means that all three will walk
away with a fabulous prize if all they did was send
us a blank cassette for an entry!

We know you're a fun loving group. We know you
like to win big prizes and fame. We know you can
write excellent programs. So get the lead out and
send us some contest entries!!!

To refresh your memory, we want programs revolving
around an "Educational" Theme. They may be in Basic,
Machine Code, or anything your little heart desires.

The deadline was set at March 15, 1987, but since
we're so nice, we decided to extend it to April 15th.
Take note, however, those entrants who were good
enough to submit their entries before the first dead
line will be given extra brownie points!

Send your entries (as many as you like) on cassette
to SyncWare News Contest, P.O. Box 64, Jefferson,
NH 03583. You may be a Winner!

win ?

SyncWare News

CONTENTS

For Your Support	3
Forum	4
Timachine (2068) REVIEW...	8
Program Segmentation...	9
Lanius	
ABC-123 Part 2 (TS1000)...	10
Nachbaur	
Chroma Soft TS1000 REVIEW...	13
2068 Piano...	14
Wentworth	
Another Video Primer...	15
Nachbaur	
2068 Winkjet1 REVIEW...	18
Put MSCRIPT on EPROM...	19
Lockin	
Mini Book Review: Best of PCW Assembler Routines for Z80...	21
Basil's Compendium...	22
Wentworth	

FIRST-CLASS MAIL
U.S. POSTAGE
PAID
Jefferson, NH
Permit No. 1

H.H. Armstrong
5301 Porter Ranch Rd
Garden Valley, CA

95633

5:2

Editor:
Jeffrey Moore
602 S. Mill Street
Louisville, OH 44641

Technical Director:
Fred Nachbaur
C-12, Mtn. Stn. Group Box
Nelson, BC V1L 5P1 Canada

Advertising, Subscriptions:
Thomas B. Woods
P.O. Box 64
Jefferson, NH 03583

Submissions, announcements, letters to the editor,
and all other material of editorial interest should
be sent to the Ohio address.

Copyright 1986 SyncWare News

Subscribe ! \$ 16.95

For Canada and Mexico please add \$3.00 postage.
All other foreign add \$9.00 Airmail (U.S. funds only)

1 year (6 issues)